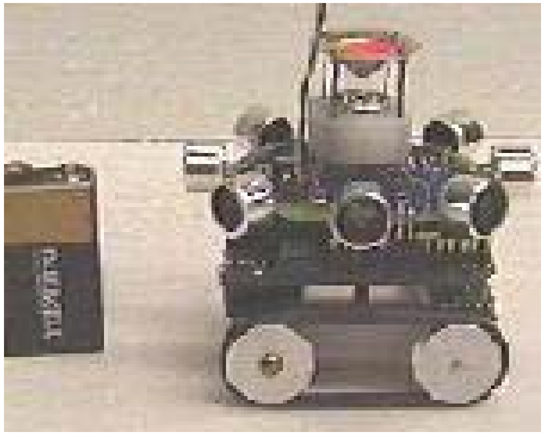
Sensing / Perception

October 3, 2002

Class Meetings 12-13



Schedule Reminder

- Today: Makeup class for Tuesday, 10/1:
  1. Meeting until 6:10
  2. Or, Friday, 10/4, 9:00 – 10:00, room 223

- Remember: Assignment #3 due at beginning of next class (Oct. 8)

Today's Objectives

- Understand various definitions related to sensing/perception
- Understand variety of sensing techniques
- Understand challenges of sensing and perception in robotics

“Old View” of Perception vs. “New View”

- Traditional (“old view”) approach:
  - Perception considered in isolation (i.e., disembodied)
  - Perception “as king” (e.g., computer vision is “the” problem)
  - Universal reconstruction (i.e., 3D world models)

"New View" of Perception

Perception without the context of action is meaningless.

- Action-oriented perception
  - Perceptual processing tuned to meet motor activities' needs
- Expectation-based perception
  - Knowledge of world can constrain interpretation of what is present in world
- Focus-of-attention methods
  - Knowledge can constrain where things may appear in the world
- Active perception
  - Agent can use motor control to enhance perceptual processing via sensor positioning
- Perceptual classes:
  - Partition world into various categories of potential interaction

Consequence of “New View”

- Purpose of perception is motor control, not representations
- Multiple parallel processes that fit robot’s different behavioral needs are used
- Highly specialized perceptual algorithms extract necessary information and no more.

Perception is conducted on a “need-to-know” basis

Complexity Analysis of New Approach is Convincing

- Bottom-up "general visual search task" where matching is entirely data driven:
  - Shown to be NP-complete (i.e., computationally intractable)

- Task-directed visual search:
  - Has linear-time complexity (Tsotsos 1989)
  - Tractability results from optimizing the available resources dedicated to perceptual processing (e.g., using attentional mechanisms)

- *Significance of results for behavior-based robotics cannot be understated:*
  - *"Any behaviorist approach to vision or robotics must deal with the inherent computational complexity of the perception problem: otherwise the claim that those approaches scale up to human-line behavior is easily refuted." (Tsotsos 1992, p. 140)*

[Tsotsos, 1989] J. Tsotsos, "The Complexity of Perceptual Search Tasks", Proc. of Int'l. Joint Conf. On Artificial Intelligence '89, Detroit, MI, pp. 1571-77.

[Tsotsos, 1992] J. Tsotsos, "On the Relative Complexity of Active versus Passive Visual Search", *International Journal of Computer Vision*, 7 (2): 127-141.

Primary Purpose of Perceptual Algorithms…

… is to support particular behavioral needs

- Directly analogous with general results we've discussed earlier regarding "hierarchical" robotic control vs. "behavior-based/reactive" robotic control

“Open-loop” vs. “Closed-loop” Control

- Closed-loop control system: Uses sensory feedback from results of its output to help compute subsequent controller outputs

- Open-loop control system: Does not use sensory feedback to evaluate the results of its actions

Sensing/Perception Definitions

- Sensor: Device that measures some attribute of the world
- Transducer: Mechanism that transforms the energy associated with what is being measured into another form of energy
  - Often used interchangeably with "sensor"
- Passive sensor: relies on environment to provide medium/energy for observation (e.g., ambient light for computer vision)
- Active sensor: puts out energy into the environment to either change energy or enhance it (e.g., laser in a laser range scanner)
- Active sensing: system for using an effector to dynamically position a sensor for a "better look"
- "Active sensor" ≠ "Active sensing":

Sensor Modalities

- Sensor modality:
  - Sensors which measure same form of energy and process it in similar ways
  - “Modality” refers to the raw input used by the sensors

- Different modalities:
  - Sound
  - Pressure
  - Temperature
  - Light
    - Visible light
    - Infrared light
    - X-rays
    - Etc.

Logical Sensors

- Logical sensor:
  - Unit of sensing or module that supplies a particular percept
  - Consists of: signal processing from physical sensor, plus software processing needed to extract the percept
  - Can be easily implemented as a perceptual schema

- Logical sensor contains all available alternative methods of obtaining a particular percept
  - Example: to obtain a 360^{o} polar plot of range data, can use:
    - Sonar
    - Laser
    - Stereo vision
    - Texture
    - Etc.

Logical Sensors (con't.)

- Logical sensors can be used interchangeably if they return the same percept
- However, not necessarily equivalent in performance or update rate
- Logical sensors very useful for building-block effect -- recursive, reusable, modular, etc.

Behavioral Sensor Fusion

- Sensor suite: set of sensors for a particular robot
- Sensor fusion: any process that combines information from multiple sensors into a single percept
- Multiple sensors used when:
  - A particular sensor is too imprecise or noisy to give reliable data
- Sensor reliability problems:
  - False positive:
    - Sensor leads robot to believe a percept is present when it isn't
  - False negative:
    - Sensor causes robot to miss a percept that is actually present

Three Types of Multiple Sensor Combinations

1. Redundant (or, competing)
   - Sensors return the same percept
   - Physical vs. logical redundancy:
     - Physical redundancy:
       - Multiple copies of same type of sensor
       - Example: two rings of sonar placed at different heights
     - Logical redundancy:
       - Return identical percepts, but use different modalities or processing algorithms
       - Example: range from stereo cameras vs. laser range finder

Three Types of Multiple Sensor Combinations (con't.)

2. Complementary
   – Sensors provide disjoint types of information about a percept
   – Example: thermal sensor for detecting heat + camera for detecting motion

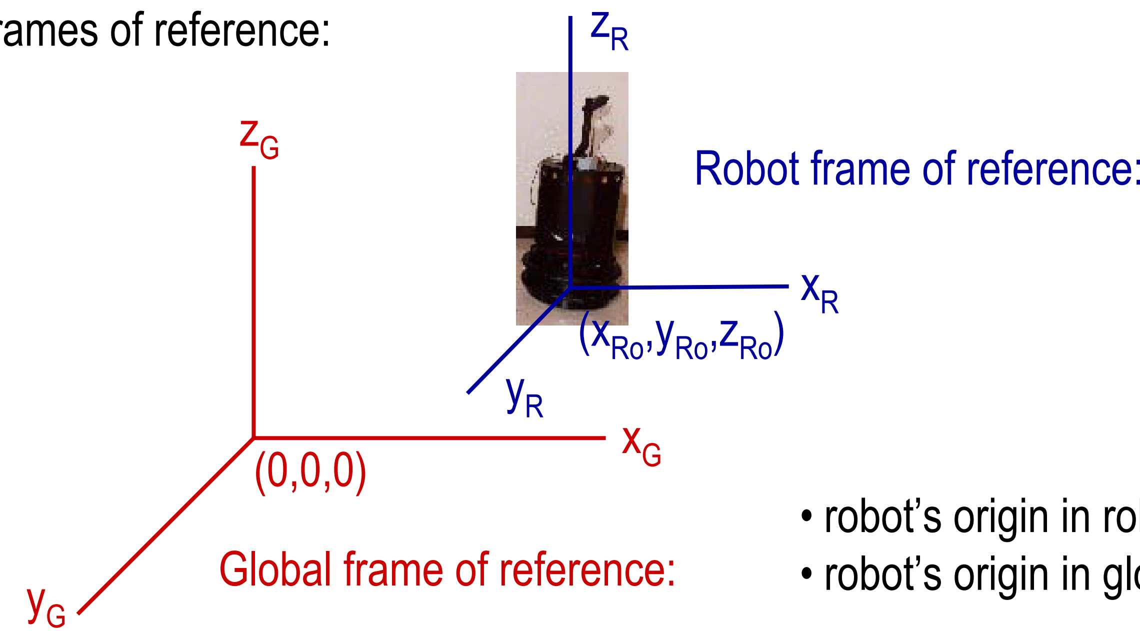
3. Coordinated
   – Use a sequence of sensors
   – Example: cue-ing or focus-of-attention; see motion, then activate more specialized sensor

Categorizing Perceptual Stimuli

- Proprioception: measurements of movement relative to the robot's internal frame of reference (also called *dead reckoning*)
- Exteroception: measurements of layout of the environment and objects relative to robot's frame of reference
- Exproprioception: measurement of the position of the robot body or parts relative to the layout of the environment

---

Frames of reference:



- robot's origin in robot's frame of reference = (0,0,0)
- robot's origin in global frame of reference = (x_{Ro},y_{Ro},z_{Ro})

Physical Attributes of a Sensor

- Field of view (FOV) and range
  - FOV usually expressed in degrees
  - Can have different horizontal and vertical FOVs
  - Critical to matching a sensor to an application
- Accuracy, repeatability, and resolution
  - Accuracy: how correct the sensor reading is
  - Repeatability: how consistent the measurements are in the same circumstances
  - Resolution: granularity of result (e.g., 1 m resolution vs. 1 cm resolution)
- Responsiveness in the target domain
  - Environment must allow the signal of interest to be extracted
  - Need favorable signal-to-noise ratio

Physical Attributes of a Sensor (con't.)

- Power consumption
  - On-board robot battery supplies limit power availability for sensors
  - Large power consumption less desirable
  - Generally, passive sensors have less power demands than active sensors
- Hardware reliability
  - Physical limitations may constrain performance (e.g., due to moisture, temperature, input voltage, etc.)
- Size
  - Has to match payload and power capabilities of robot

Computability Attributes of a Sensor

- Computational complexity
  - Estimate of how may operations the sensor processing algorithm requires
  - Serious problem for smaller robot vehicles
- Interpretation reliability
  - Software interpretation issues
  - Difficulty of interpreting sensor readings
  - Difficulty of recognizing sensor errors

Selecting Appropriate Sensor Suite

Desired attributes of entire sensory suite:

- Simplicity
- Modularity
- Redundancy (enables fault tolerance)
  - Physical
  - Logical

Today: Overview of Common Sensors for Robotics

- Note: our overview will be from the software functionality level
- For more hardware-related implementation details, see:
  - *Sensors for Mobile Robots*, by H. R. Everett, published A K Peters Ltd, 1995.
- Keep in mind:
  - All of these sensors have a variety of hardware implementations
  - Many hardware details affect capability and performance of sensors
- We won't be discussing hardware design issues beyond general level of concept understanding

Major Categories of Sensors

- Proprioceptive
- Proximity
- Computer vision
- Mission-specific

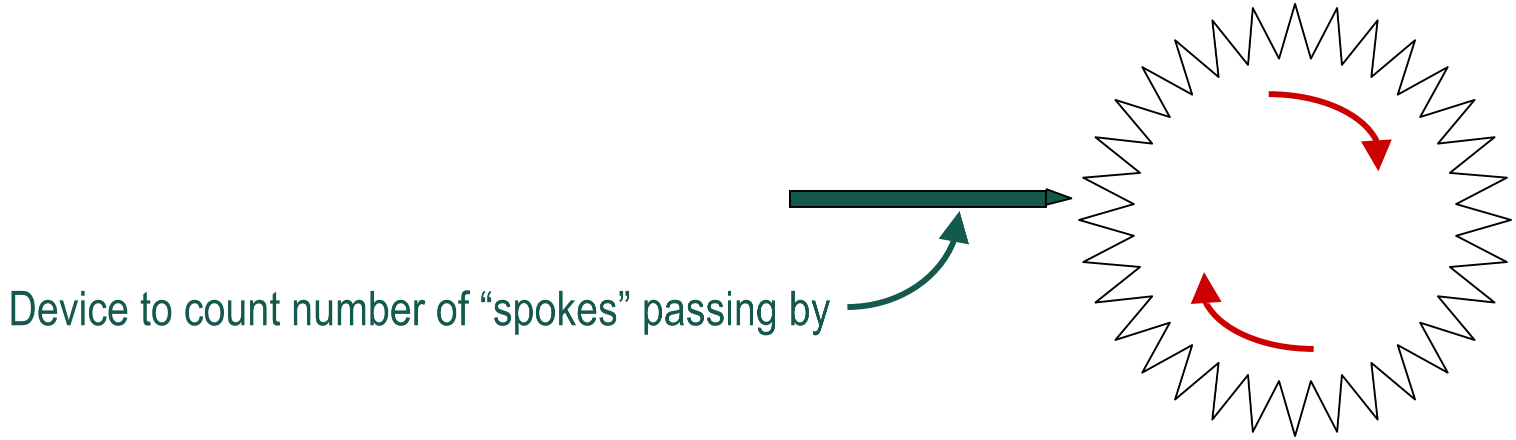
Proprioceptive Sensors

- Sensors that give information on the internal state of the robot, such as:
  - Motion
  - Position (x, y, z)
  - Orientation (about x, y, z axes)
  - Velocity, acceleration
  - Temperature
  - Battery level
- Example proprioceptive sensors:
  - Encoders (dead reckoning)
  - Inertial navigation system (INS)
  - Global positioning system (GPS)
  - Compass
  - Gyroscopes

Dead Reckoning/Odometry/Encoders

- Purpose:
  - To measure turning distance of motors (in terms of numbers of rotations), which can be converted to robot translation/rotation distance
- If gearing and wheel size known, number of motor turns ➔ number of wheel turns ➔ estimation of distance robot has traveled

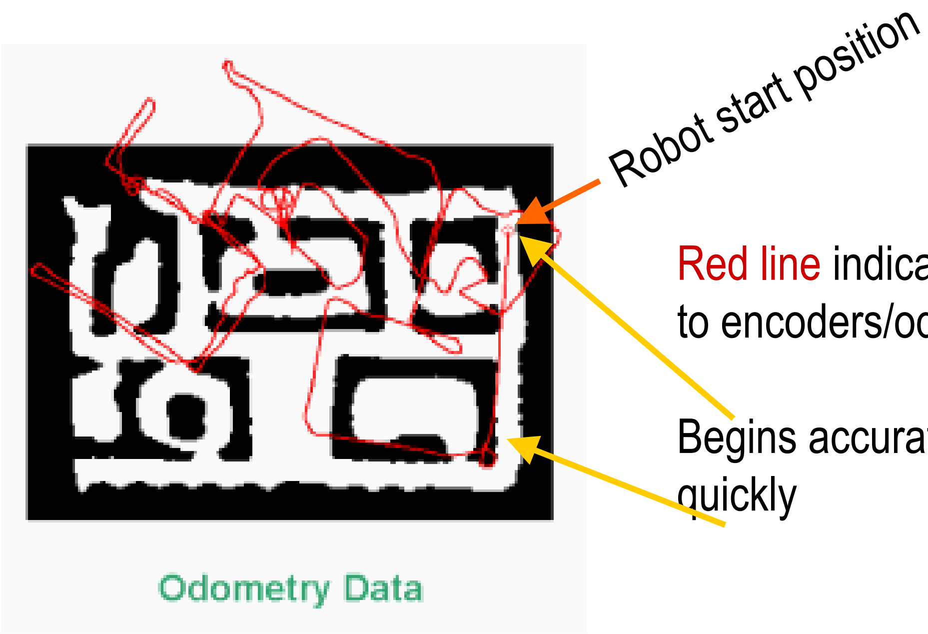
- Basic idea in hardware implementation:



Encoders (con't.)

- Challenges/issues:
  - Motion of wheels not corresponding to robot motion, e.g., due to wheel spinning
  - Wheels don't move but robot does, e.g., due to robot sliding

- Error accumulates quickly, especially due to turning:

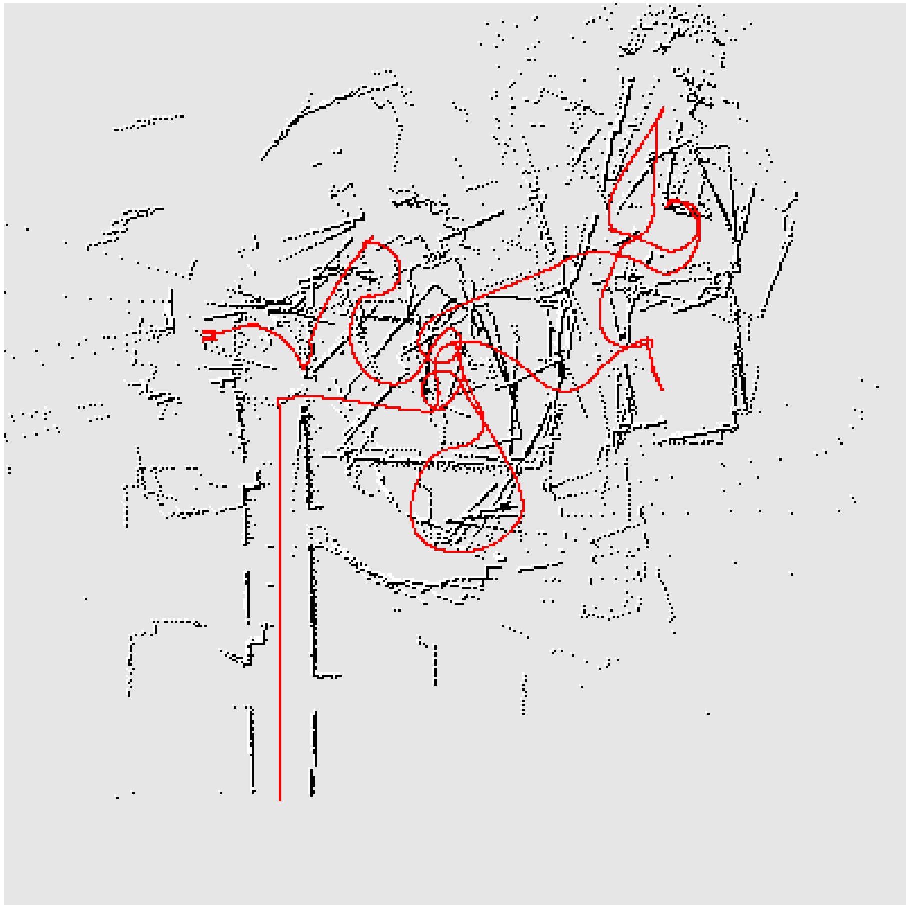


Red line indicates estimated robot position due to encoders/odometry/dead reckoning.

Begins accurately, but errors accumulate quickly

Another Example of Extent of Dead Reckoning Errors

- Plot of overlaid laser scans overlaid based strictly on odometry:



Inertial Navigation Sensors (INS)

- Inertial navigation sensors: measure movements electronically through miniature accelerometers

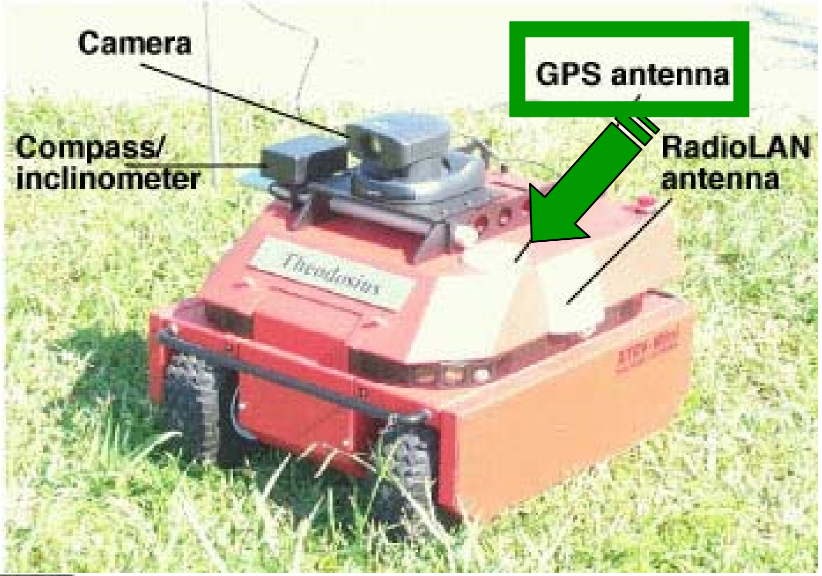
- Accuracy: quite good (e.g., 0.1% of distance traveled) if movements are smooth and sampling rate is high

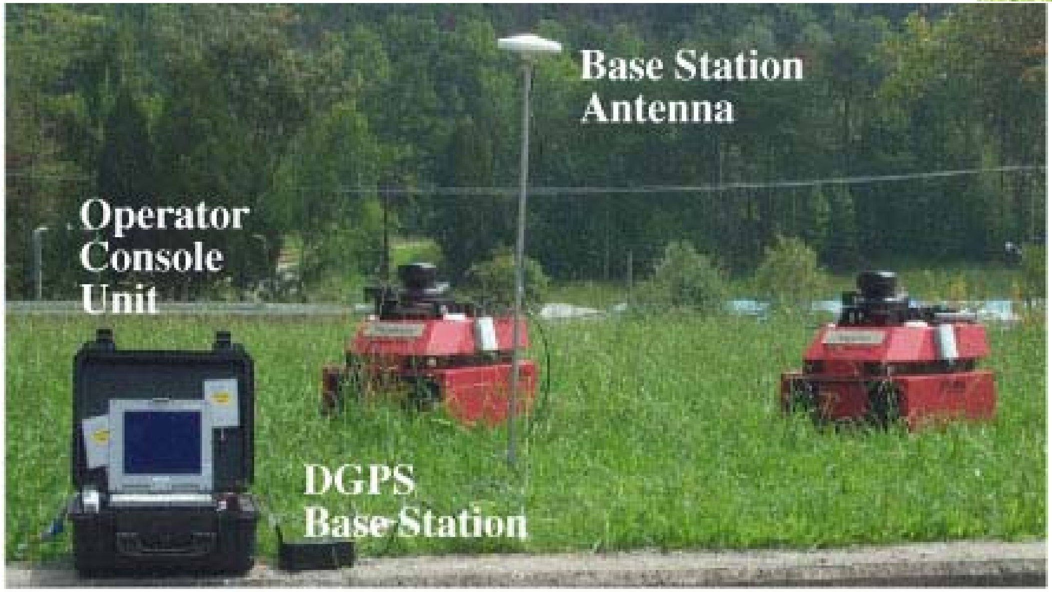
- Problem for mobile robots:
  - Expensive: $50,000 - $100,000 USD
  - Robots often violate smooth motion constraint
  - INS units typically large

Differential Global Positioning System (DGPS)

- Satellite-based sensing system
- Robot GPS receiver:
  - Triangulates relative to signals from 4 satellites
  - Outputs position in terms of latitude, longitude, altitude, and change in time
- Differential GPS:
  - Improves localization by using two GPS receivers
  - One receiver remains stationary, other is on robot
- Sensor Resolution:
  - GPS alone: 10-15 meters
  - DGPS: up to a few centimeters

Example DGPS Sensors on Robots





DGPS Challenges

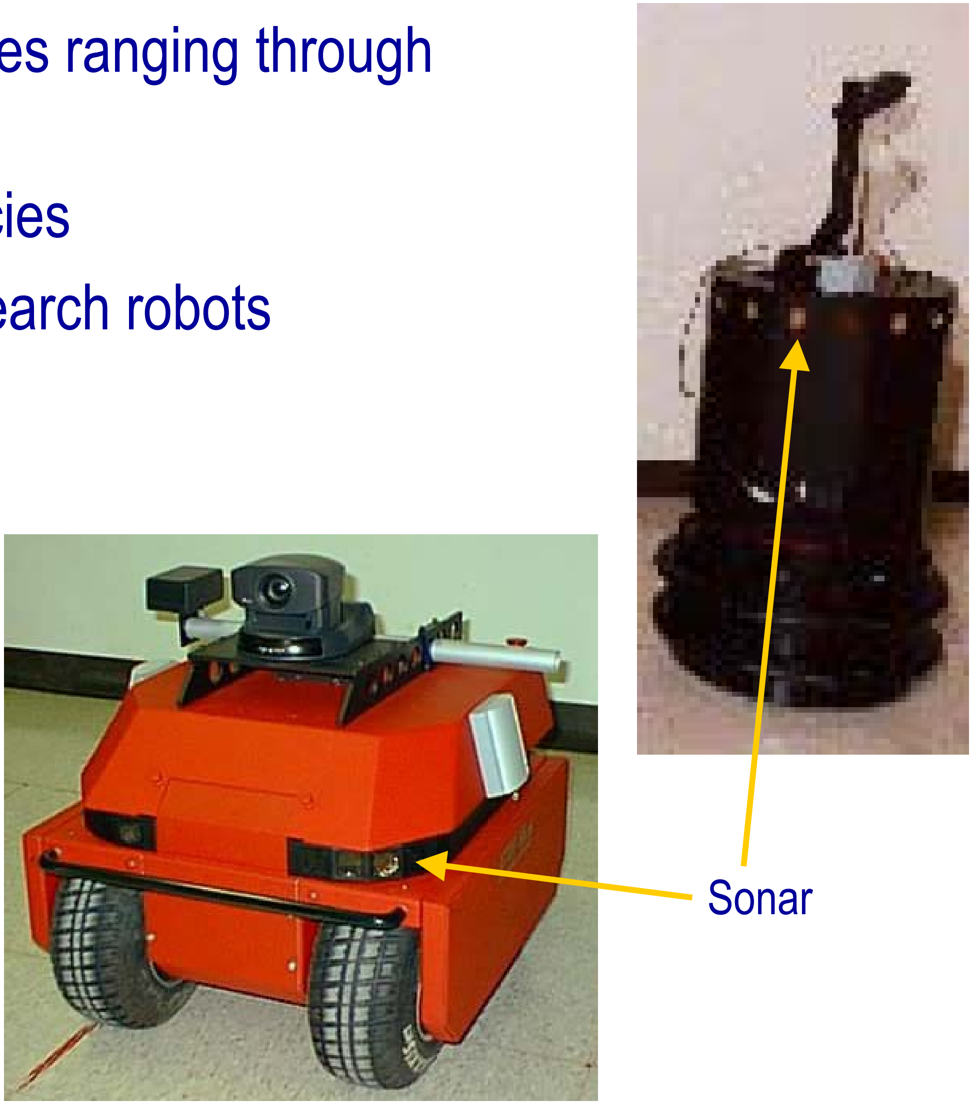
- Does not work indoors in most buildings
- Does not work outdoors in “urban canyons” (amidst tall buildings)
- Forested areas (i.e., trees) can block satellite signals
- Cost is high (about $30,000)

Proximity Sensors

- Measure relative distance (range) between sensor and objects in environment
- Most proximity sensors are active
- Common Types:
  - Sonar (ultrasonics)
  - Infrared (IR)
  - Bump and feeler sensors

Sonar (Ultrasonics)

- Refers to any system that achieves ranging through sound
- Can operate at different frequencies
- Very common on indoor and research robots
- Operation:
  - Emit a sound
  - Measure time it takes for sound to return
  - Compute range based on *time of flight*

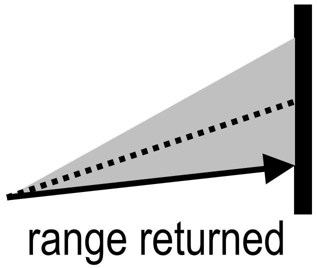


Reasons Sonar is So Common

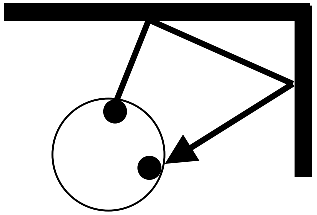
- Can typically give 360^o coverage as polar plot
- Cheap (a few $US)
- Fast (sub-second measurement time)
- Good range – about 25 feet with 1” resolution over FOV of 30^o

Sonar Challenges

- “Dead zone”, causing inability to sense objects within about 11 inches
- Indoor range (up to 25 feet) better than outdoor range (perhaps 8 feet)
- Key issues:
  - Foreshortening:

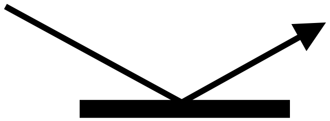


  - Cross-talk: sonar cannot tell if the signal it is receiving was generated by itself, or by another sonar in the ring



Sonar Challenges (con't.)

- Key issues (con't.)
  - Specular reflection: when wave form hits a surface at an acute and bounces away



  - Specular reflection also results in signal reflecting differently from different materials
    - E.g., cloth, sheetrock, glass, metal, etc.

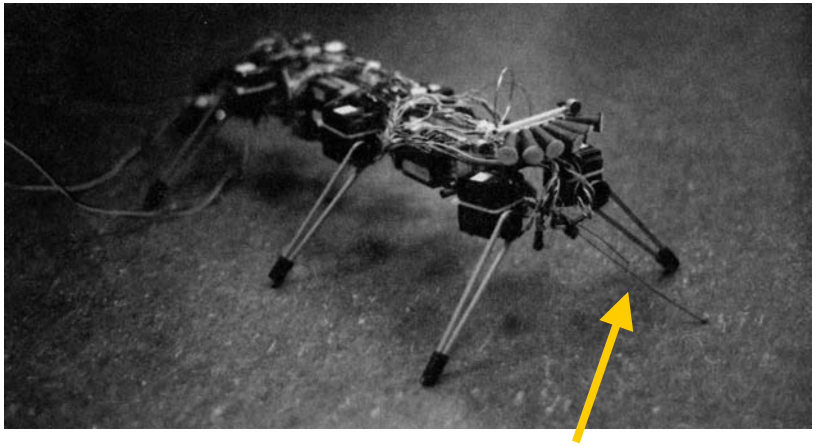
- Common method of dealing with spurious readings:
  - Average three readings (current plus last two) from each sensor

Infrared (IR)

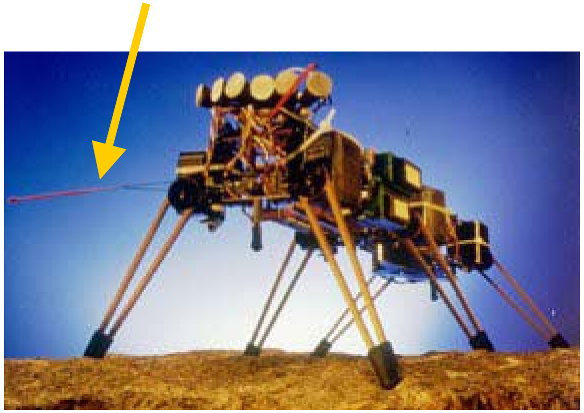
- Active proximity sensor
- Emit near-infrared energy and measure amount of IR light returned
- Range: inches to several feet, depending on light frequency and receiver sensitivity
- Typical IR: constructed from LEDs, which have a range of 3-5 inches
- Issues:
  - Light can be “washed out” by bright ambient lighting
  - Light can be absorbed by dark materials

Bump and Feeler (Tactile) Sensors

- Tactile (touch) sensors: wired so that when robot touches object, electrical signal is generated using a binary switch
- Sensitivity can be tuned ("light" vs. "heavy" touch), although it is tricky
- Placement is important (height, angular placement)



Whiskers on Genghis



Computer Vision

- Computer vision: processing data from any modality that uses the electromagnetic spectrum which produces an image
- Image:
  - A way of representing data in a picture-like format where there is a direct physical correspondence to the scene being imaged
  - Results in a 2D array or grid of readings
  - Every element in array maps onto a small region of space
  - Elements in image array are called pixels
- Modality determines what image measures:
  - Visible light → measures value of light (e.g. color or gray level)
  - Thermal → measures heat in the given region
- Image function: converts signal into a pixel value

Types of Computer Vision

- Computer vision includes:
  - Cameras (produce images over same electromagnetic spectrum that humans see)
  - Thermal sensors
  - X-rays
  - Laser range finders
  - Synthetic aperature radar

Computer Vision is a Field of Study on its Own

- Computer vision field has developed algorithms for:
  - Noise filtering
  - Compensating for illumination problems
  - Enhancing images
  - Finding lines
  - Matching lines to models
  - Extracting shapes and building 3D representations

- *However, behavior-based/reactive robots tend not to use these algorithms, due to high computational complexity*

CCD (Charge Couple Device) Cameras

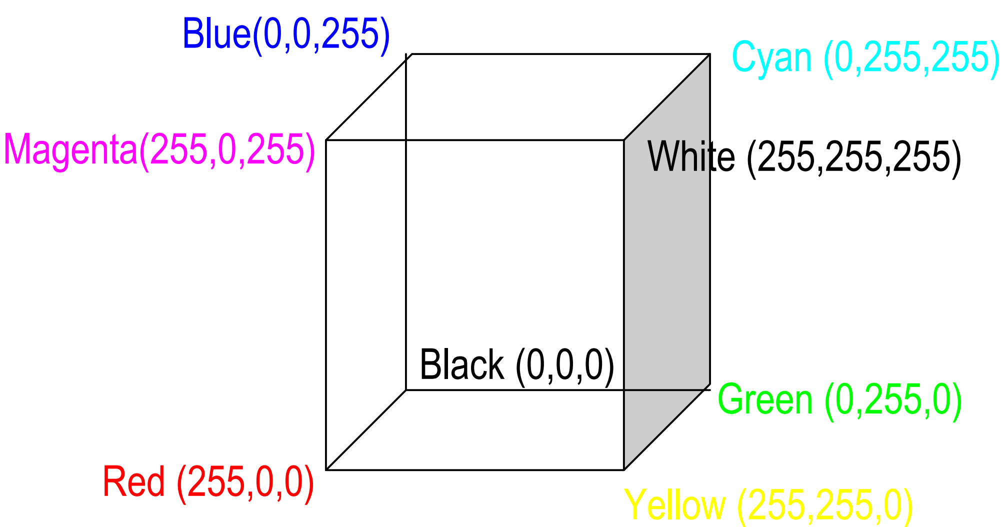
- CCD technology: Typically, computer vision on reactive/behavior-based robots is from a video camera, which uses CCD technology to detect visible light

- Output of most cameras: analog; therefore, must be digitized for computer use

- Framegrabber:
  - Card that is used by the computer, which accepts an analog camera signal and outputs the digitized results
  - Can produce gray-scale or color digital image
  - Have become fairly cheap – color framegrabbers cost about $300-$500.

Representation of Color

- Color measurements expressed as three color planes – red, green, blue (abbreviated RGB)
- RGB usually represented as axes of 3D cube, with values ranging from 0 to 255 for each axis



Software Representation

1. Interleaved: colors are stored together (most common representation)
   - Order: usually red, then green, then blue

Example code:

```
#define RED 0
#define GREEN 1
#define BLUE 2

int  image[ROW][COLUMN][COLOR_PLANE];
…
red = image[row][col][RED];
green = image[row][col][GREEN];
blue = image[row][col][BLUE];
display_color(red, green, blue);
```

Software Representation (con't.)

2. Separate: colors are stored as 3 separate 2D arrays

Example code:

```
int   image_red[ROW][COLUMN];
int   image_green[ROW][COLUMN];
int   image_blue[ROW][COLUMN];

…
red = image_red[row][col];
green = image_green[row][col];
blue = image_blue[row][col];
display_color(red, green, blue);
```

Challenges Using RGB for Robotics

- Color is function of:
  - Wavelength of light source
  - Surface reflectance
  - Sensitivity of sensor

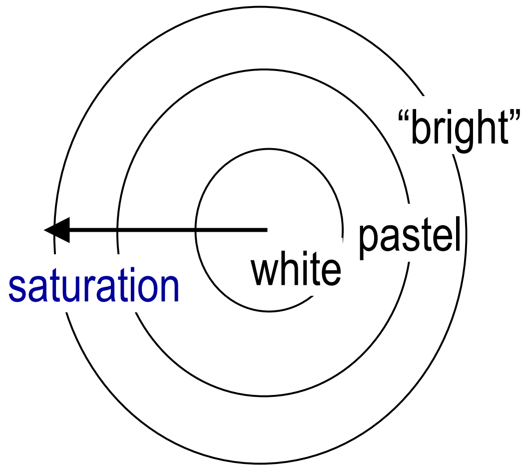
- → Color is not absolue;
  - Object may appear to be at different color values at different distances to due intensity of reflected light

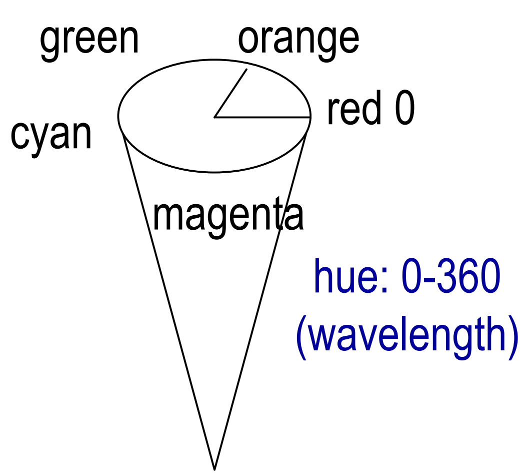
Better: Device which is sensitive to absolute wavelength

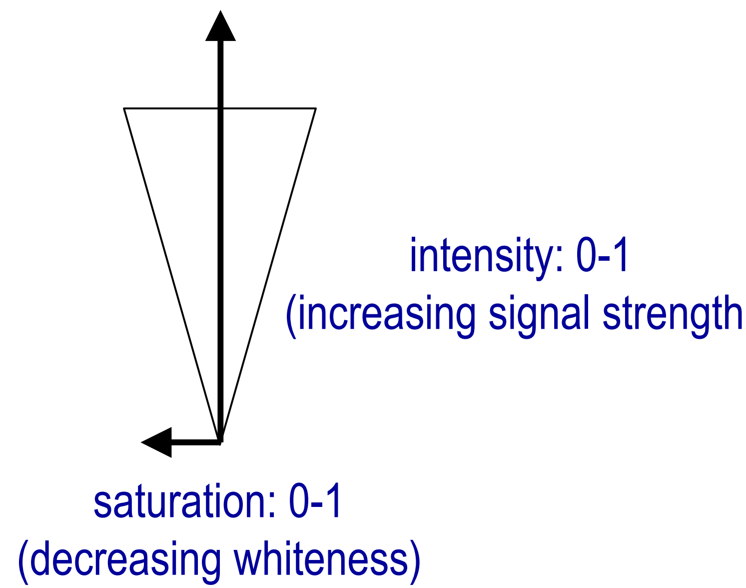
Better: Hue, saturation, intensity (or value) (HSV) representation of color

- Hue: dominant wavelength, does not change with robot's relative position or object's shape
- Saturation: lack of whiteness in the color (e.g., red is saturated, pink is less saturated)
- Intensity/Value: quantity of light received by the sensor

Representation of HSV







HSV Challenges for Robotics

- Requires special cameras and framegrabbers
- Very expensive equipment

- Alternative: Spherical Coordinate Transform (SCT)
  - Transforms RGB data to a color space that more closely duplicates response of human eye
  - Used in biomedical imaging, but not widely used for robotics
  - Much more insensitive to lighting changes

Region Segmentation

- Region Segmentation: most common use of computer vision in robotics, with goal to identify region in image with a particular color

- Basic concept: identify all pixels in image which are part of the region, then navigate to the region's centroid
- Steps:
  - Threshold all pixels which share same color (thresholding)
  - Group those together, throwing out any that don't seem to be in same area as majority of the pixels (region growing)

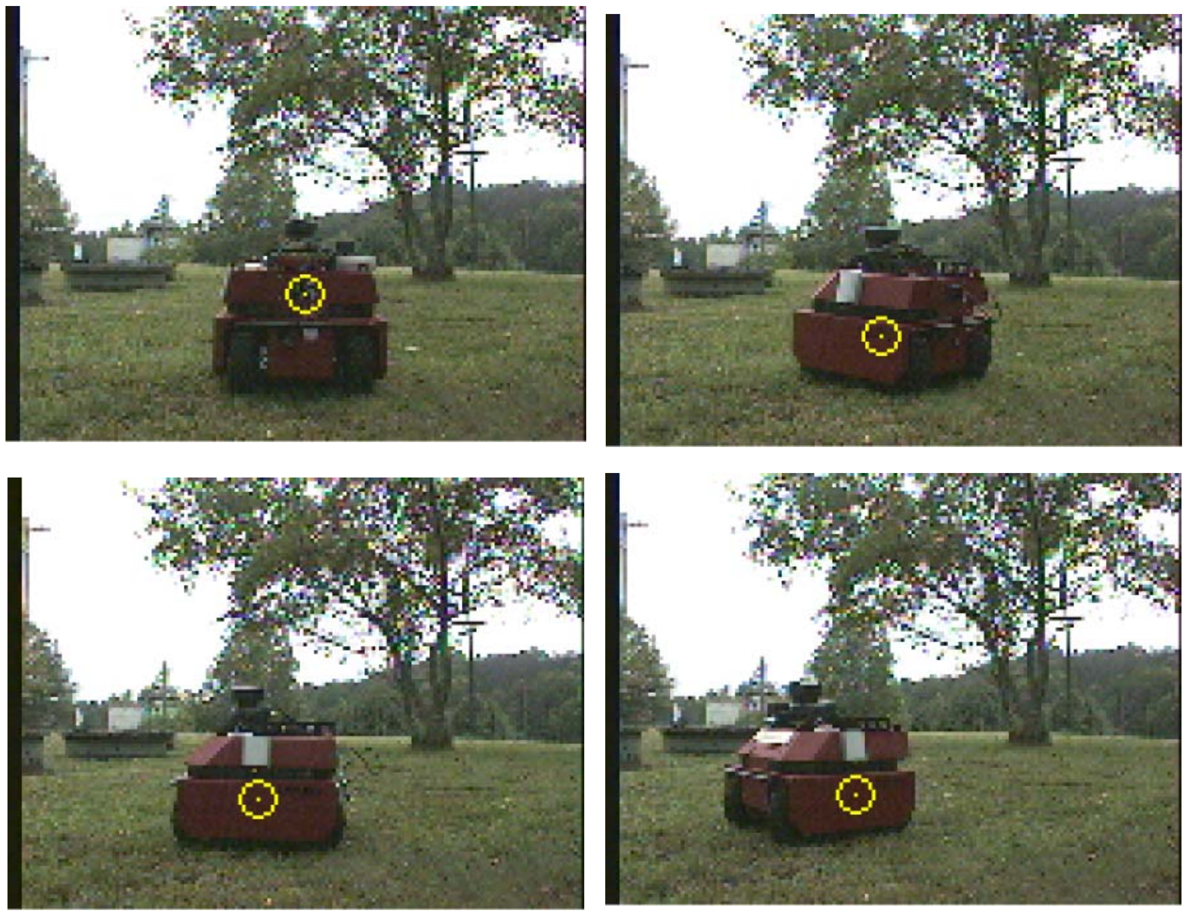
Example Code for Region Segmentation

```
for (i=0; i<numberRows; i++)
  for (j=0; j<numberColumns; j++)
    { if (((ImageIn[i][j][RED] >= redValueLow)
        && (ImageIn[i][j][RED] <= redValueHigh))
        && ((ImageIn[i][j][GREEN] >= greenValueLow)
        && (ImageIn[i][j][GREEN] <= greenValueHigh))
        && ((ImageIn[i][j][BLUE] >= blueValueLow)
        && (ImageIn[i][j][BLUE] <= blueValueHigh)))
        ImageOUT[i][j] = 255;
      else
        ImageOut[i][j] = 0;
    }
```

Note range of readings required due to non-absolute color values

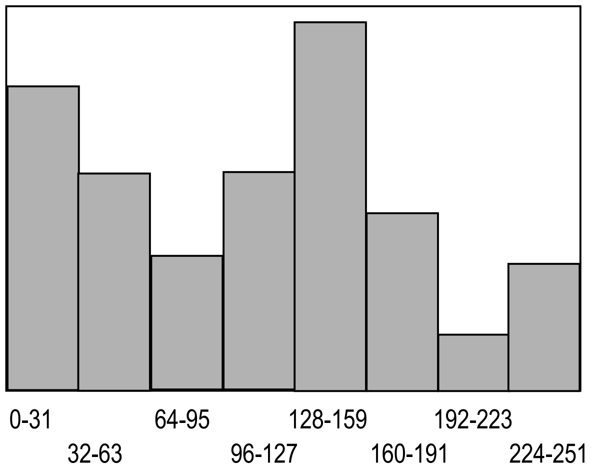
Example of Region-Based Robotic Tracking using Vision

Another Example of Vision-Based Robot Detection Using Region Segmentation



Color Histogramming

- Color histogramming:
  - Used to identify a region with several colors
  - Way of matching proportion of colors in a region
- Histogram:
  - Bar chart of data
  - User specifies range of values for each bar (called buckets)
  - Size of bar is number of data points whose value falls into the range for that bucket
- Example:

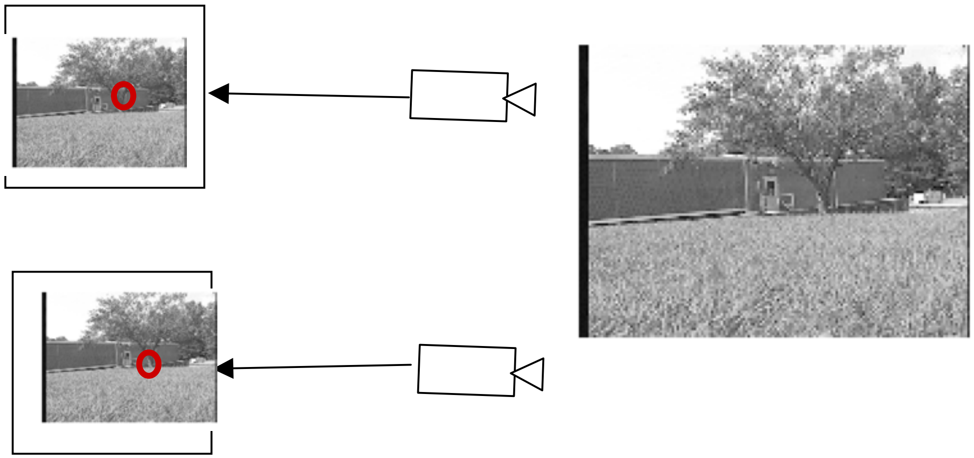


Color Histograms (con't.)

- Advantage for behavior-based/reactive robots: Histogram Intersection
  - Color histograms can be subtracted from each other to determine if current image matches a previously constructed histogram
  - Subtract histograms bucket by bucket; different indicates # of pixels that didn't match
  - Number of mismatched pixels divided by number of pixels in image gives percentage match = Histogram Intersection
- Useful for detecting affordances
- This is example of local, behavior-specific representation that can be directly extracted from environment

Range from Vision

- Perception of depth from stereo image pairs, or from optic flow
- Stereo camera pairs: range from stereo
- Key challenge: how does a robot know it is looking at the same point in two images?
  - This is the correspondence problem.

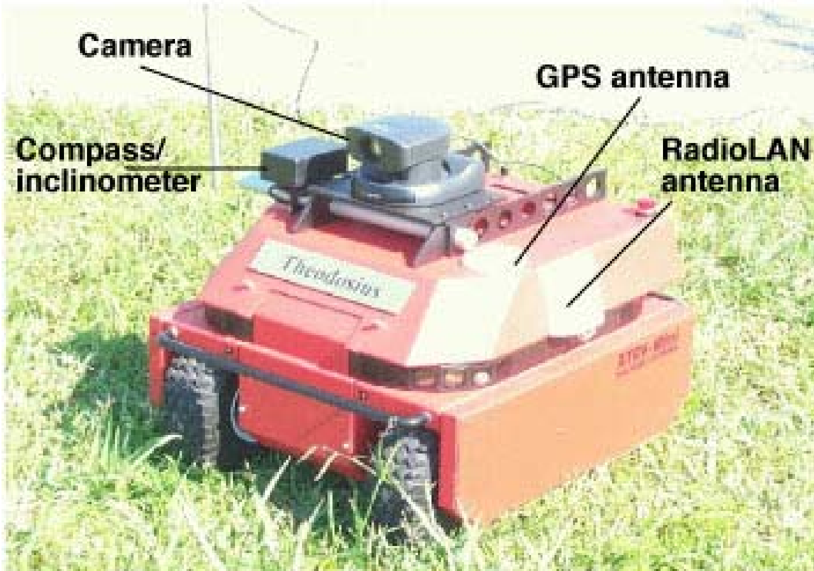


Simplified Approach for Stereo Vision

- Given scene and two images
- Find interest points in one image
- Compute matching between images (correspondence)
- Distance between points of interest in image is called disparity
- Distance of point from the cameras is inversely proportional to disparity
- Use triangulation and standard geometry to compute depth map

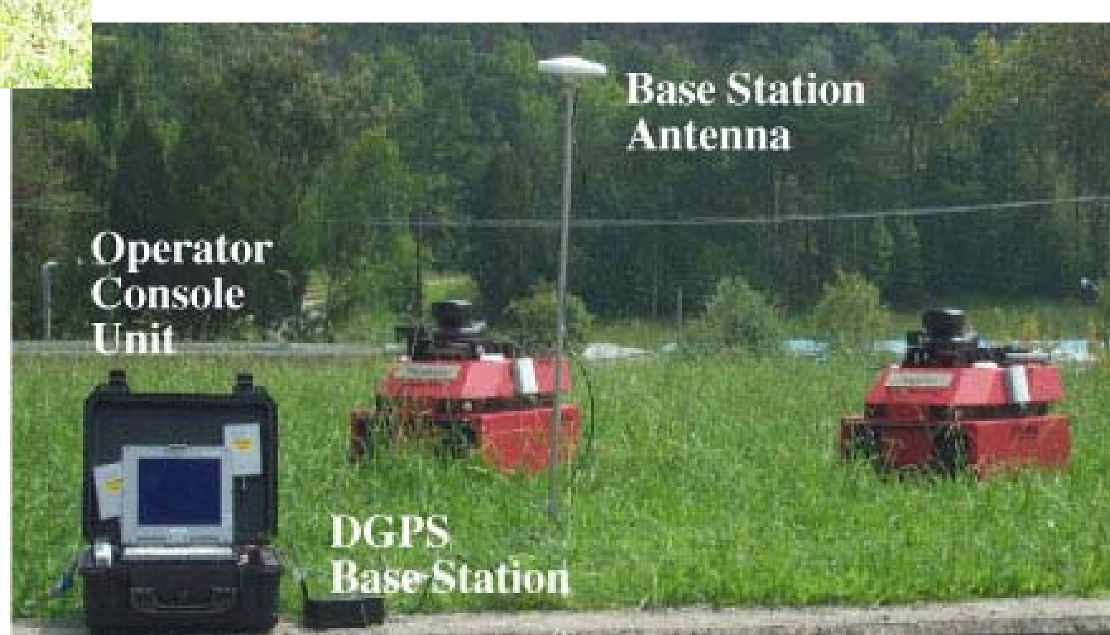
- Issue: camera calibration: need known information on relative alignment between cameras for stereo vision to work properly

Example of Computing Depth from Multiple Images



- Robot Team: 4 ATRV-mini robots (Manuf: RWI/iRobot)
  - Named (after Roman Emperors): Augustus, Constantine, Theodosius, Vespasian

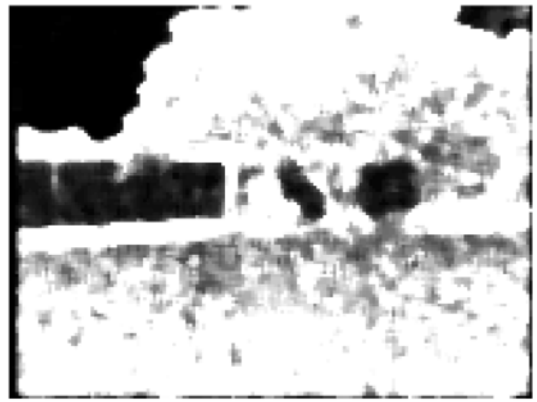
- Sensors:
  - 2 robots: PTZ camera
  - 2 robots: SICK laser
  - Compass/inclinometer
  - DGPS
  - Sonar

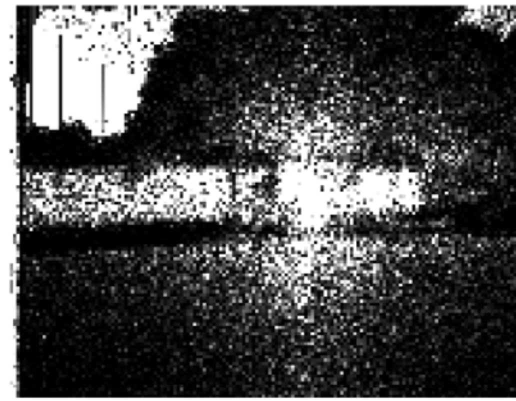


Example Results of Depth Maps

Actual scene | Depth map | Depth covariance

Augustus:





Theodosius:

Preview of Next Class (Tuesday, Oct. 8th)

- More about Computer Vision robotic applications
- Conclusions of Sensing/Perception
- Representational Issues for Behavior-Based Robotics