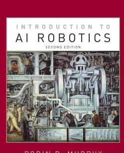
11

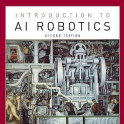
# Range Sensing



11

# Objectives

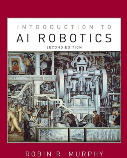
- Describe the problems of specular reflection, cross-talk, and foreshortening with an ultrasonic transducer, and, if given a 2D line drawing of surfaces, illustrate where each of these problems would likely occur.
- Describe the point cloud knowledge representation and the types of algorithms associated with it.
- Describe the two popular types of active range sensors for producing point clouds: lidar and structured light
- Be familiar with Sense-Register-Reconstruct sequence



11

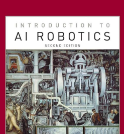
# Overview

- The terms “depth” and “range”
  - the distance between the robot and a surface in the world.
  - But, high resolution sensors that can be used to determine the three-dimensional shape of an object are generally called **depth sensors** to differentiate those sensors from **range sensors**.
- Range can be sensed with either passive or active sensors
  - Passive: triangulation of features seen in two images or from cues in the image.
  - Active sensors, such as sonars and lidar, send a pulse of sound or light, respectively, into the environment and measure the time of flight for the signal to return.



# Overview (Cont.)

- Point cloud
  - A new knowledge structure (range map).
  - Created from using high-resolution range sensors.
  - Similar with an occupancy grid
  - But, the term point cloud is now dominating.
- RGB-D systems, where D stands for depth
  - Microsoft's Kinect
  - The Google Tango project
- This chapter covers the major classes of algorithms and knowledge representations for determining the range to an object or surface



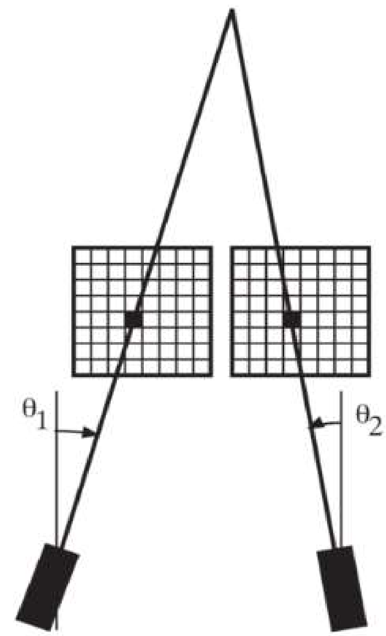
# Range from Vision

- Perception of depth from stereo image pairs, or from optic flow
- Stereo camera pairs: range from stereo

Try to superimpose a feature point over each camera. Each camera finds the same point in each image, turns itself to center that point in the image, and then the relative angle is measured.

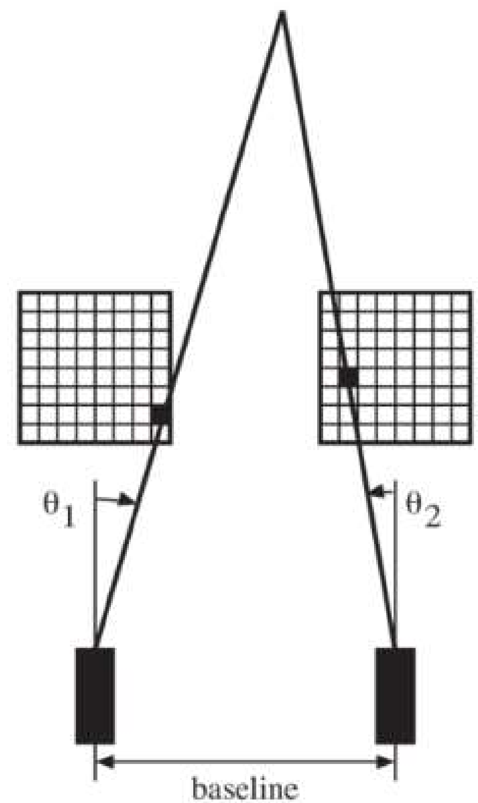
Key challenge: how does a robot know it is looking at the same point in two images?

This is the correspondence problem.

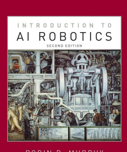


# Simplified Approach for Stereo Vision

- Given scene and two images
- Find **interest points** in one image
  - very bright or dark spots or edges
- 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




The algorithm that selects interesting points is called an **interest operator**



# Advantages and Disadvantages

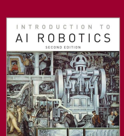


- Advantages
  - Passive
  - Good coverage
- Disadvantages
  - Environment may not support interest operators
  - Computationally expensive
  - Sensitive to camera calibration
    - Alignment between cameras for stereo vision
  - Sensitive to lighting

# Depth from X



- About them
  - Usually single camera, single image or short image sequence
  - Useful cues about the depth include
    - Motion → Depth from motion
    - Focal length → Depth from focus
    - Change of texture → Depth from texture
    - Location of shadows → Depth from shading
- Advantages
  - Nice theory, often computationally optimized
  - Usually have a camera anyway
- Disadvantages
  - Tend to be brittle outside of highly controlled conditions



# 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*

11

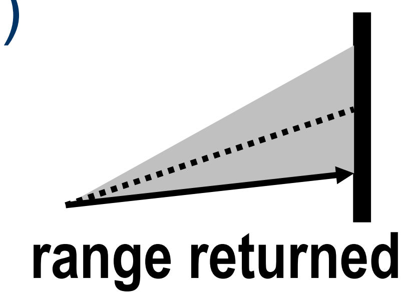
# Reasons Sonar is So Common During the 1980s and 1990s

- A ring of sonars 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 $30^{o}$

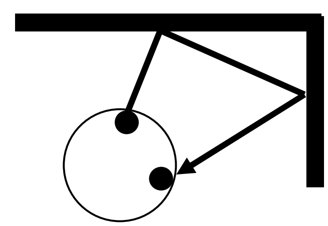
11

# 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:

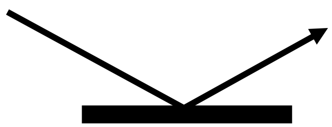



  - Cross-talk: sonar cannot tell if the signal it is receiving was generated by itself, or by another sonar in the ring



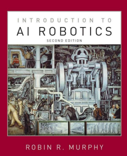
# Sonar Challenges (cont.)

- Key issues (cont.)
  - Specular reflection: when waveform hits a surface at an acute and bounces away



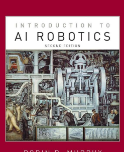
  - Specular reflection also results in signal reflecting differently from different materials
    - E.g., cloth, sheetrock, glass, metal, etc.

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

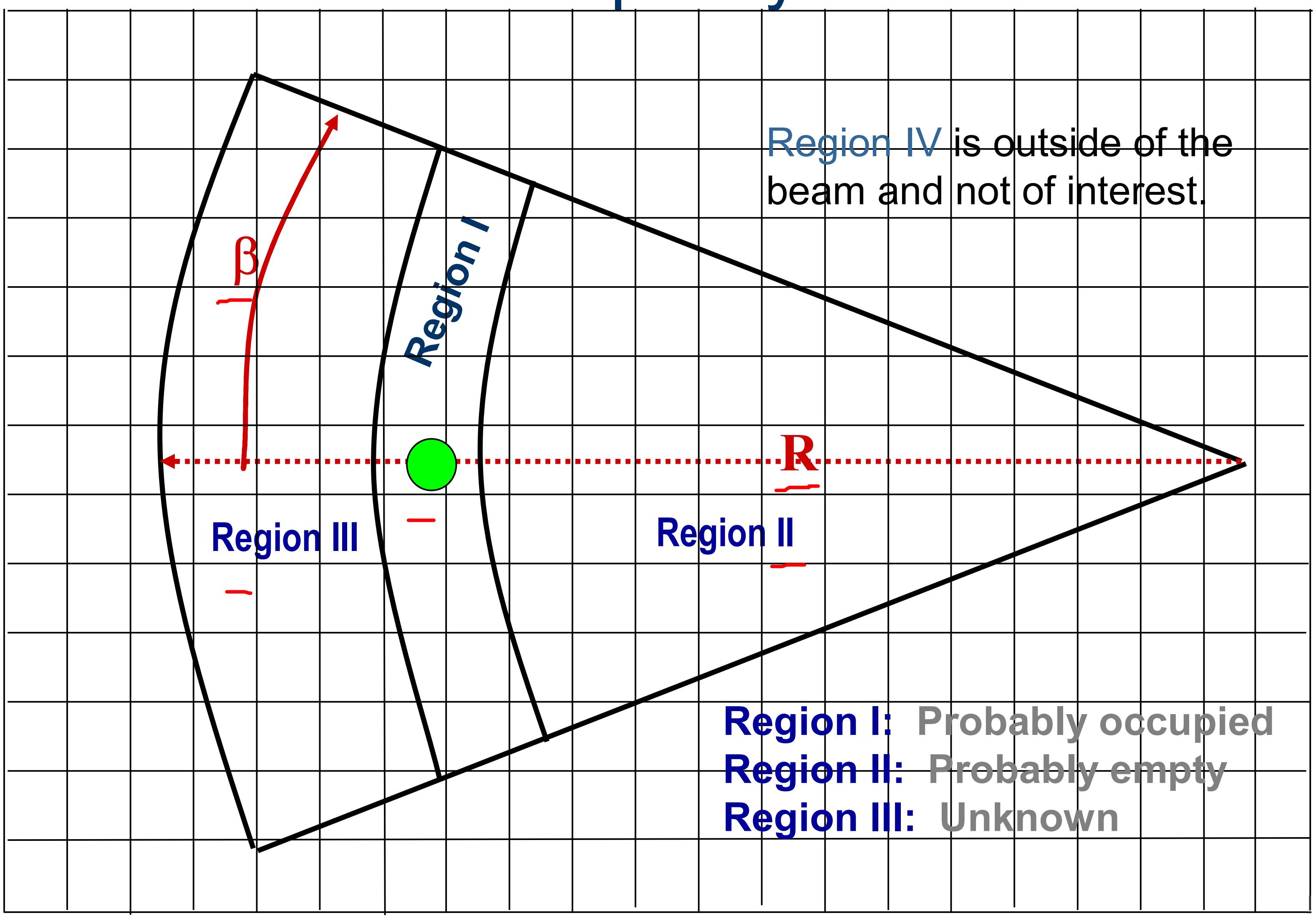


# 11 Sonar Sensor Model and Occupancy Grid

- Need sensor model to deal with uncertainty
- Methods for generating sensor models:
  - Empirical (i.e., through testing)
  - Analytical (i.e, through understanding of physical properties)
  - Subjective (i.e., through experience)
- Idea: Create *occupancy grids* to improve certainty, then treat occupancy grid as the sensor output (virtual sensor)
- Note: The following content regarding to modeling sonar sensor and updating occupancy grid are from the first edition of the textbook (Chapter 11).



# Modeling Common Sonar Sensor Occupancy Grid



11

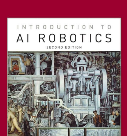
# How to Convert to Numerical Values?

- Need to translate model (previous slide) to specific numerical values for each occupancy grid cell
- Bayesian Approach:
  - Convert sensor readings into probabilities
  - Combine probabilities using Bayes' rule:

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

  - That is,

$$\text{Posterior} = \frac{\text{Likelihood} \times \text{Prior}}{\text{Normalizing constant}}$$

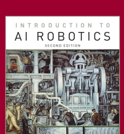


# 11 Review: Basic Probability Theory

- Probability function:
  – Gives values from 0 to 1 indicating whether a particular event H has occurred
- For sonar sensing:
  – Event: Sensing out acoustic wave and measuring time of flight
  – Outcome: Range reading reporting whether the region being sensed is Occupied or Empty
- H = {Occupied}, or {Empty}
- Probability that H has really occurred:

  $0 \leq P(H) \leq 1$
- Probability that H has not occurred:

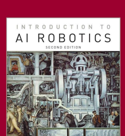
  $1 - P(H)$



11

# Unconditional and Conditional Probabilities

- Unconditional probability: $P(H)$
  - "Probability of *H*"
  - Only provide *a priori* information
  - For example, could give the known distribution of rocks in the environment, e.g., "x% of environment is covered by rocks"
  - For robotics, unconditional probabilities are not based on sensor readings
- Conditional probability: $P(H \mid s)$
  - "Probability of *H*, given *s*"
  - For robotics, based on sensor readings, *s*
- Note: $P(H \mid s) + P(\text{not } H \mid s) = 1.0$



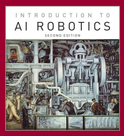
11

# Probabilities for Occupancy Grids

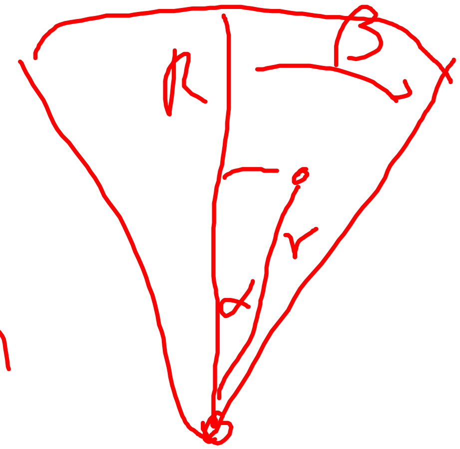
- For each grid[i][j] covered by sensor scan:
  - Compute *P*(*Occupied* | *s*) and *P*(*Empty* | *s*)
- For each grid element, grid[i][j], store tuple of the two probabilities:

```
typedef struct {
    double occupied;
    double empty;
  } P;

P occupancy_grid[ROWS][COLUMNS];
```



# Converting Sonar Reading to Probability: Region I




- Region I:

The nearer the grid element to the origin of the sonar beam, the higher the belief

The closer to the acoustic axis, the higher the belief

$$P(Occupied) = \frac{\frac{R-r}{R} + \frac{\beta - \alpha}{\beta}}{2} \times Max_{occupied}$$

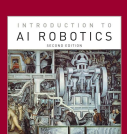
We never know with certainty

where $r$ is distance to grid element,

$\alpha$ is angle to grid element

$Max_{occupied}$ = highest probability possible (e.g., 0.98)

*P*(*Empty*) = 1.0 – *P*(*Occupied*)



# Converting Sonar Reading to Probability: Region II

- Region II:

**The nearer the grid element to the origin of the sonar beam, the higher the belief**

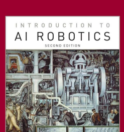
**The closer to the acoustic axis, the higher the belief**

$$P(Empty) = \frac{\frac{R - r}{R} + \frac{\beta - \alpha}{\beta}}{2}$$

$P(Occupied) = 1.0 - P(Empty)$

where $r$ is distance to grid element,

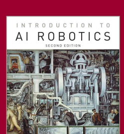
$\alpha$ is angle to grid element



11

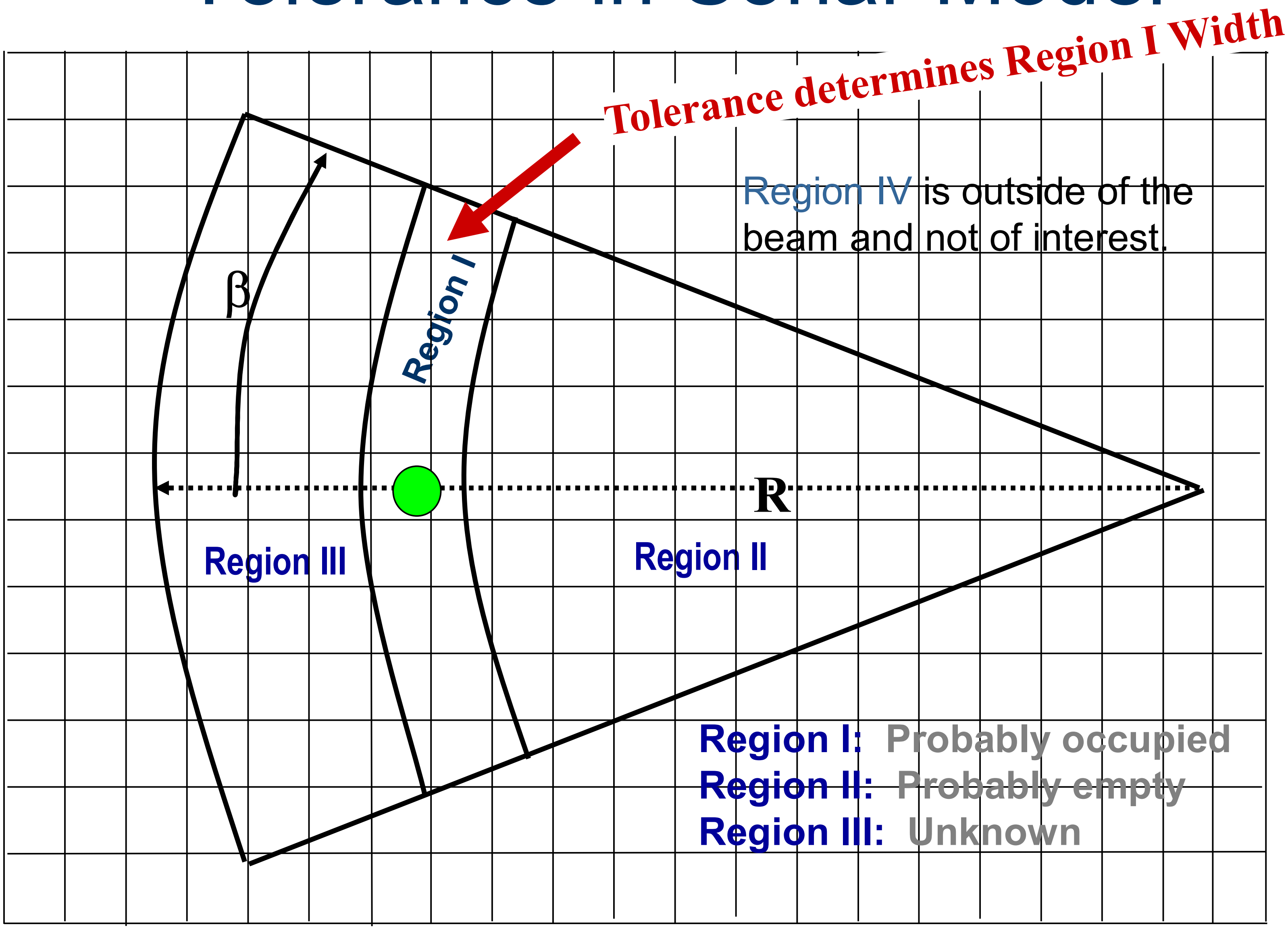
# Sonar Tolerance

- Sonar range readings have resolution error
- Thus, specific reading might actually indicate range of possible values
- E.g., reading of 0.87 meters actually means within (0.82, 0.92) meters
  - Therefore, *tolerance* in this case is 0.05 meters.
- Tolerance gives width of Region I

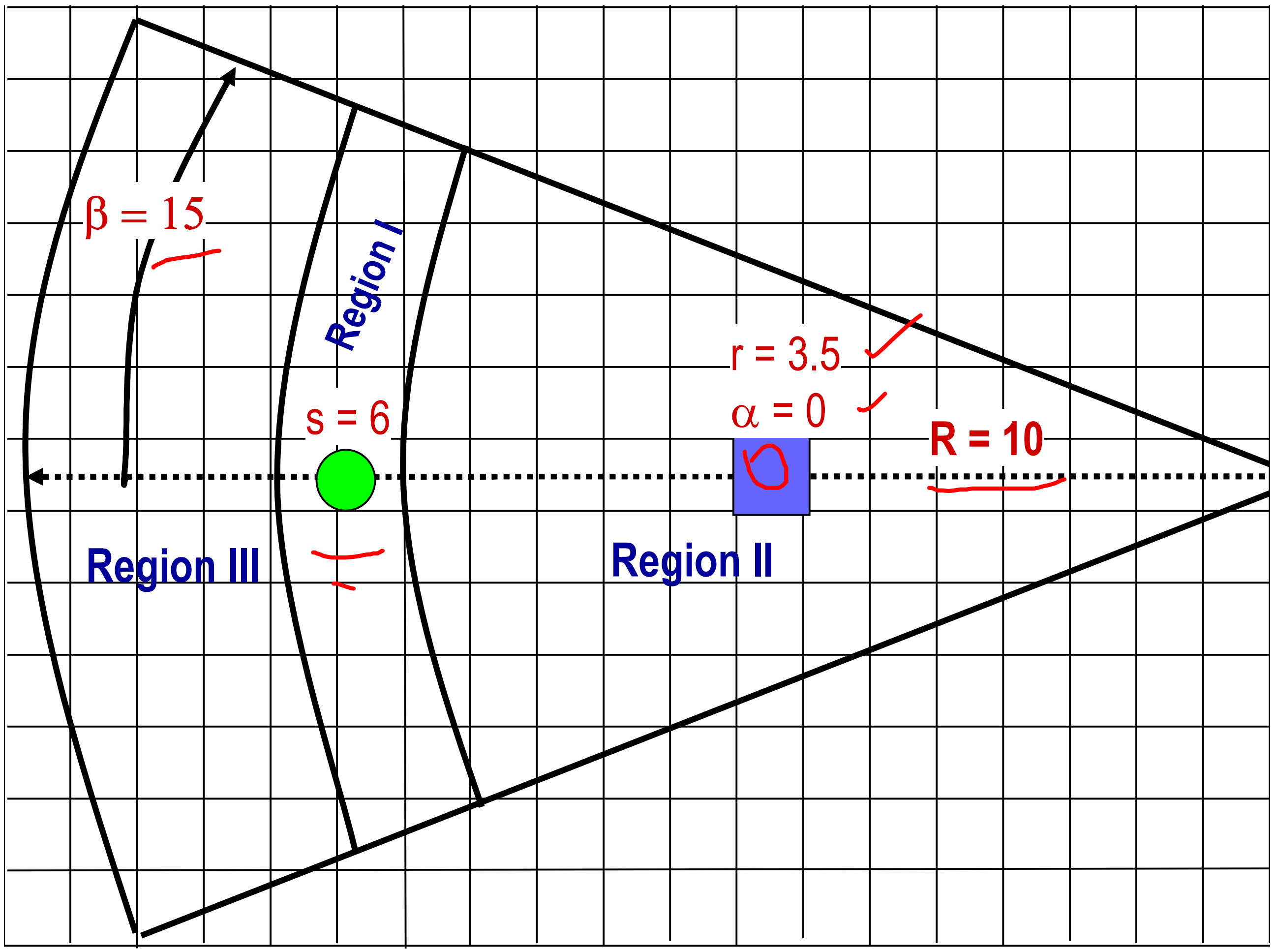


11

# Tolerance in Sonar Model



# Example: What is value of grid cell ?




Which region?

3.5 < (6.0 – 0.5) ➔ Region II

$$P(\text{Empty}) = \frac{\frac{10-3.5}{10} + \frac{15-0}{15}}{2}$$

$= 0.83$

$= P(s \mid \text{Empty})$

$P(\text{Occupied}) = (1 - 0.83) = 0.17$

$= P(s \mid \text{Occupied})$

# Returning to Example

- Let's assume we're on Mars, and we know that *P(Occupied)* = 0.75

- $P(Empty \mid s{=}6) = \dfrac{P(s \mid Empty)\, P(Empty)}{P(s \mid Empty)\, P(Empty) + P(s \mid Occupied)\, P(Occupied)}$

$$= \frac{0.83 \times 0.25}{0.83 \times 0.25 + 0.17 \times 0.75}$$

$$= 0.62$$

- $P(Occupied \mid s{=}6) = 1 - P(Empty \mid s{=}6) = 0.38$

| |
|---|
| Note: $P(s) = P(s \mid Empty)\, P(Empty) + P(s \mid Occupied)\, P(Occupied)$ |

# Updating with Bayes Rule

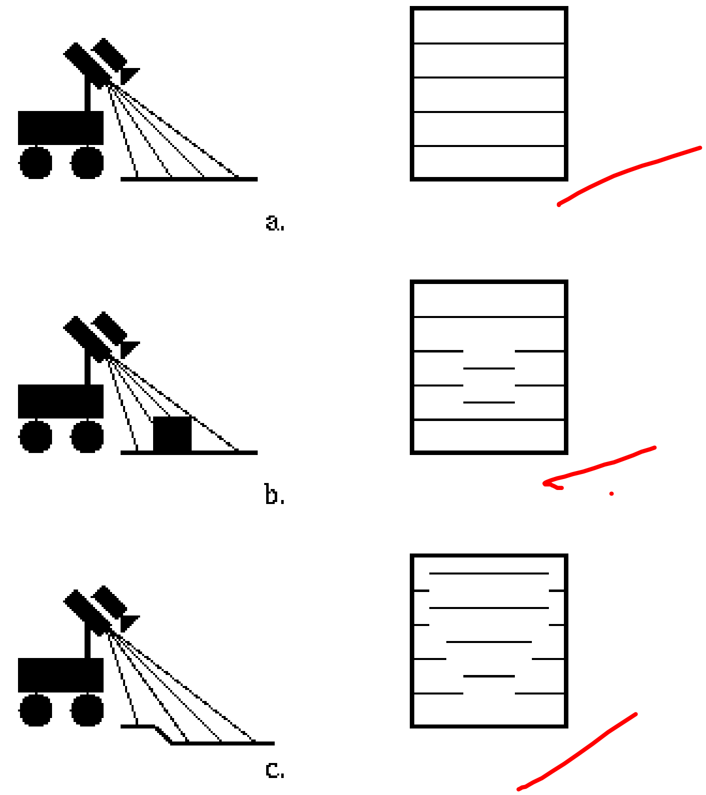
- How to fuse multiple readings?
- First time:
  - Each element of grid initialized with *a priori* probability of being occupied or empty
- Subsequently:
  - Use Bayes' rule iteratively
  - Probability at time $t_{n-1}$ becomes prior and is combined with current observation at $t_n$ :

$$P(H \mid s_n) = \frac{P(s_n \mid H)\, P(H \mid s_{n-1})}{P(s_n \mid H)\, P(H \mid s_{n-1}) + P(s_n \mid not\ H)\ P(not\ H \mid s_{n-1})}$$

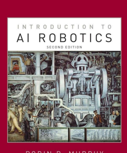
# 11

# Light Stripers

- Light stripers work by projecting a colored line (or stripe), grid, or pattern of dots on the environment.
- Then a regular video camera observers how the pattern is distorted in the image.
- A vision algorithm can quickly detect if the projected line is continuous or not
- Light stripers are less expensive.
  - Expensive time-of-flight detectors are unnecessary
  - Does not require a laser
  - Striper is less demanding, so less expensive to build

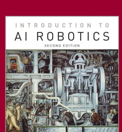



a.) flat surface
b.) an obstacle
c.) a negative obstacle (hole)



# Lidar

- Devices which use laser to produce a depth map or image are often called laser radar, or ladar, or lidar.
  - Laser radar works just like a raster scan on a CRT
- A lidar produces two images: intensity and range.
  - Intensity: measures the intensity of the light reflected or absorbed by objects in the scene.
  - Range: represents depth from the camera.

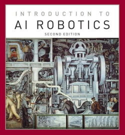


# Planar Lasers



- About them:
  - Emit a plane of eye-safe laser light, measure time of flight
  - Sick is the standard
- Advantages
  - Very accurate
  - 2D coverage
- Disadvantages
  - Expensive
  - Large
  - 2D (though can mount more, etc.)
  - See through glass

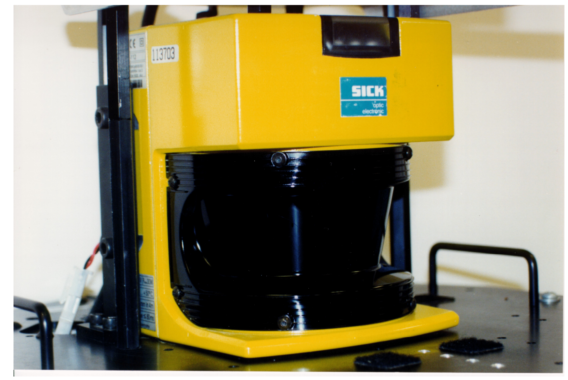
**Laser ranger**

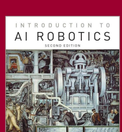


# Sick



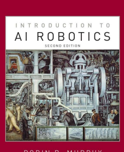
- Accuracy & repeatability
  - Excellent results
- Responsiveness in target domain
- Power consumption
  - High; reduce battery run time by half
- Reliability
  - good
- Size
  - A bit large
- Computational Complexity
  - Not bad until try to “stack up”
- Interpretation Reliability
  - Much better than any other ranger



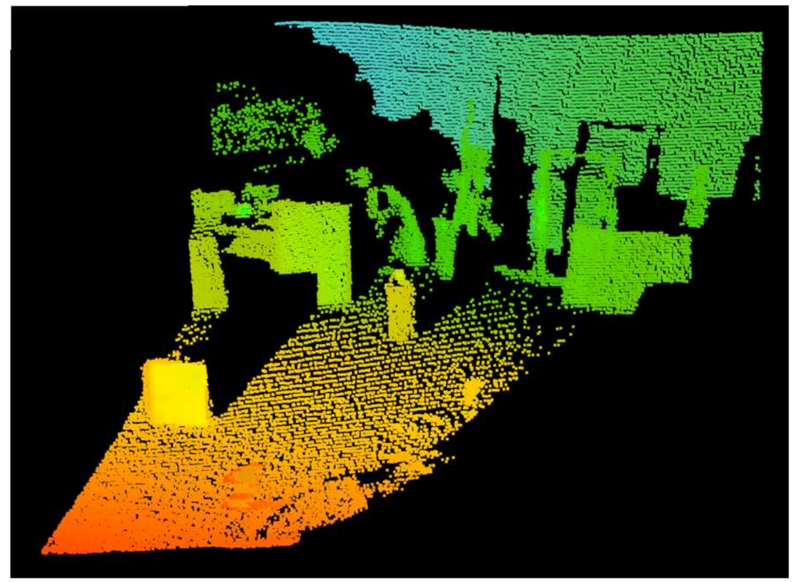
# RGB-D Cameras

- RGB-D cameras are a combination of a IR light striping system (except instead of stripes of light, random dots are projected) with a regular RGB visible light camera.
  - Microsoft's Kinect
  - The Google Tango project
  - ASUS Xtion cameras
- RGB-D cameras are not sufficient for general robotics for three reasons.
  - typically optimized for the videogame market.
  - the IR image can be noisy due to specular reflection
  - tuned for projection and receptivity over specific ranges
    - In the case of a Kinect, that range is 1.2 m to 3.5 m.

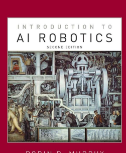


# 11 General Idea of Point Cloud

- Actively "paint" a region of space with some type of light
- Produce dense *point clouds* which support visualization, 3D reconstruction
- They are good for
  - Terrain elevation and obstacle avoidance
  - Surface reconstruction and volumetric estimation
  - Object recognition from shape and movement

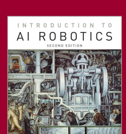


Pointclouds.org



# What are Point Clouds?

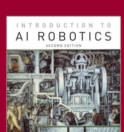
- Point clouds are a knowledge representation of the range to surfaces.
- The cloud is a collection of points.
- Each point has
  - an (x,y,z) coordinate, called a vertex, representing the distance between the sensor and the surface of an object
  - Or a vector (direction, distance) but harder to visually manipulate
- A point may also have a value such as the saturation of light



# 11 How Point Clouds Are Produced

Scan using

- Active Range Sensors
  - Lidar (laser): emit a laser beam and then record the time it takes for the beam to bounce off a surface and return.
    - Sick
    - Velodyne (scanner)
    - Hokoyu
  - Structured light sensor: emit light in near infrared frequency band that is invisible to the human eye
    - Kinect
    - LEAP
- But also from geospatial stereo imagery
- 3D scanners



11

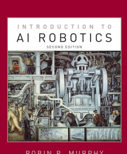
# Differences

## Active Range Sensors:

- Rapid production of points
- Slower reconstruction
- Optional/less common visual image overlay

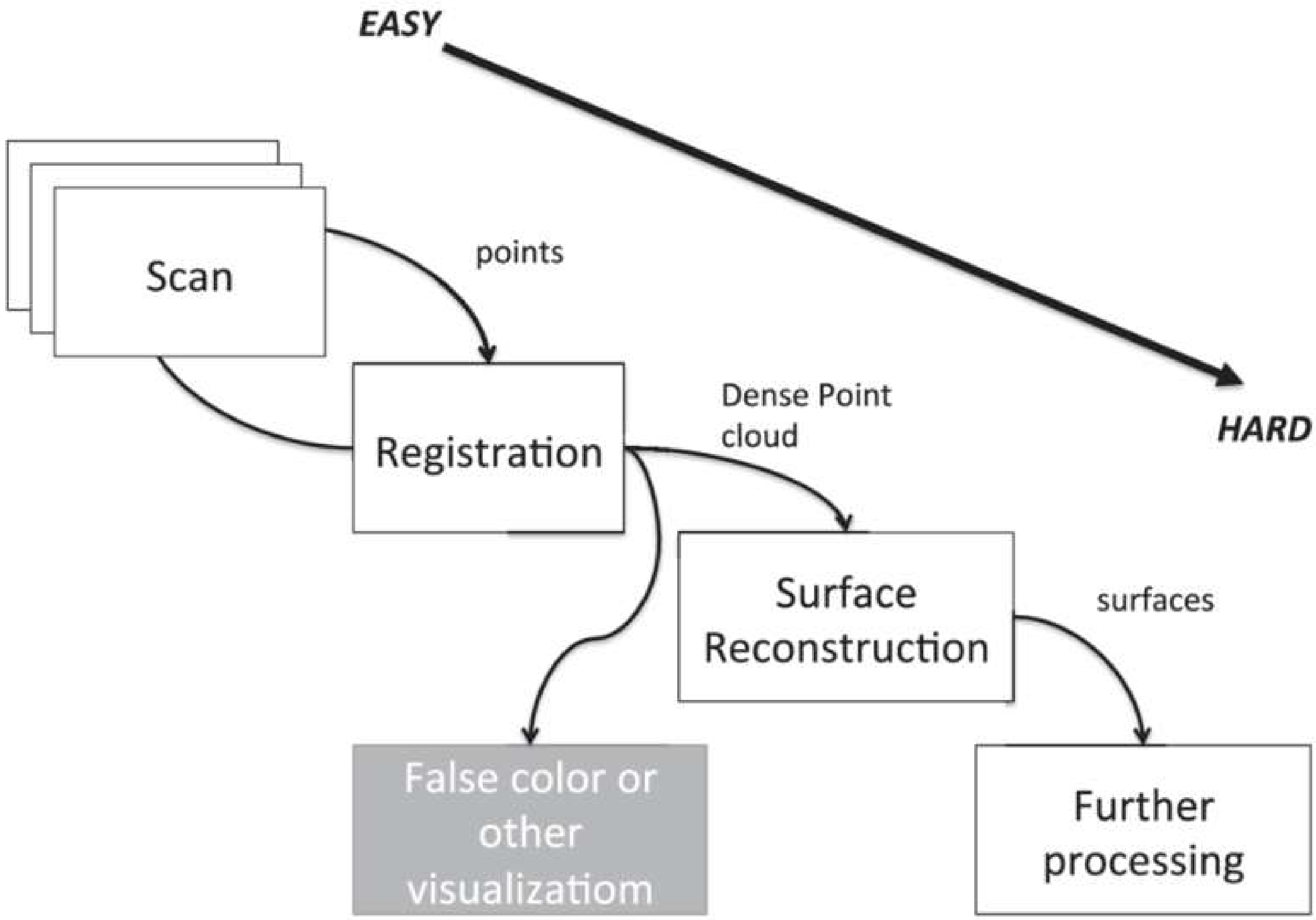
## Stereo imagery:

- Slow production of points
- Slow reconstruction
- Visual image overlay built in



11

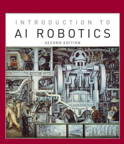
# The sequence of processing activities associated with point clouds



11

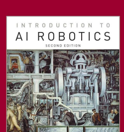
# Scan and Registration

- Scan
  - The process is for the sensor to sense the environment and generate the set of points
- Registration
  - Registration is assigning each point an x,y,z location in absolute coordinates, because the scan will be reported in relative, or egocentric, coordinates.
  - Registration can also overlay the visual image on the point cloud.
  - Must **register** the points, especially if the robot or objects were moving or multiple scans are required.



# Surface Reconstruction

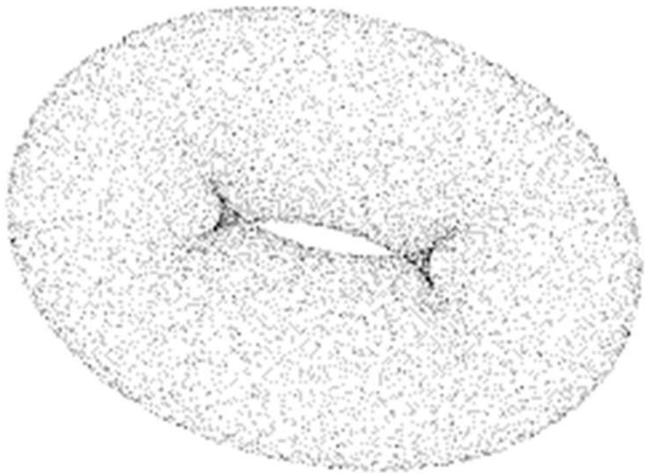
- Point clouds by themselves are often so dense that just displaying the points at different distances from the robot with false color is a reasonable visualization of the depth
- However, with more complex scenes, the false color visualization requires the human to mentally segment the world into surfaces
- Usually an additional, computationally expensive, step of processing of **surface reconstruction** for further computer processing is necessary.
  - Connect the points (vertices) into surfaces

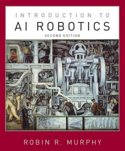


11

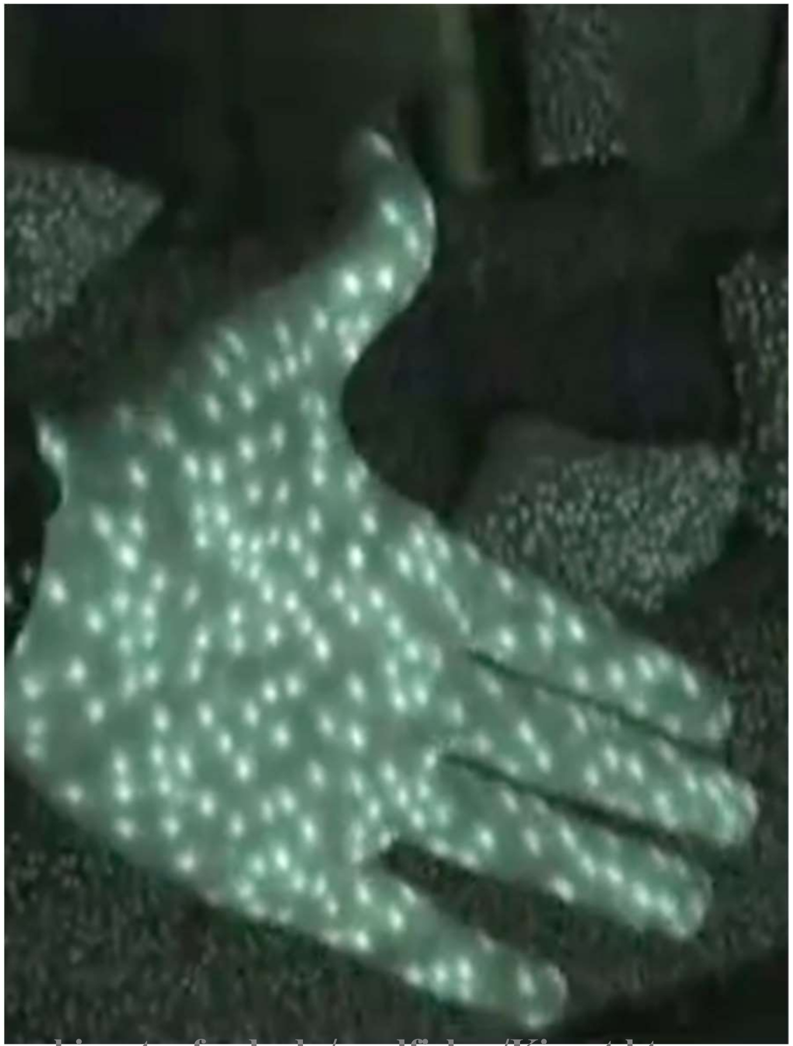
# Surface Reconstruction

- Idea: Connect the points (vertices) into surfaces
  - Mesh (polygon or triangle)
  - Non-uniform rational basis spline (curves)
- Common Methods
  - Triangle mesh (Delaunay triangulations)
  - Marching cube (look at 8 neighbors)

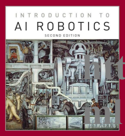




# 11 About Structured Light



- Project light at invisible frequencies
- Frequencies are generally narrow and thus can be washed out
- The sensors aren't particularly good, it's the clever post-processing and fusion with other sensors




https://graphics.stanford.edu/~mdfisher/Kinect.ht

# 11 Skeletal Tracking is Major Advance