Sensing, Yet More:
Time of Flight

Correspondence

2

- Two cameras see slightly different scenes
  - What points in one correspond to points in the other?
  - Compare all points in image to all points in other image
  - This image search can be computationally expensive, imperfect

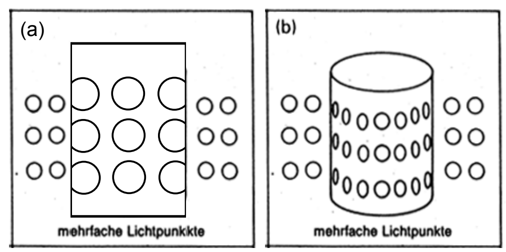
Disparity

3

Structured Light

5

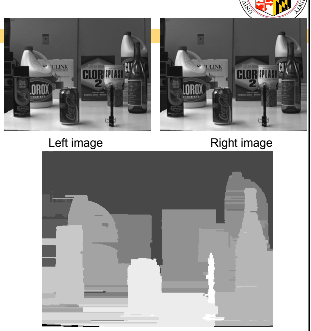
- Light is distorted by object it is falling on
- Two kinds of distortion: size and shape



Disparity Map

6

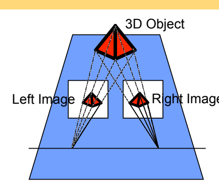
- **Distance map**: the farther away something is, the darker it is
  - For any distance sensor!
- **Disparity map**: distance map, built from both kinds of disparity



Distance images usually visualized as grey-scale images. Objects that are closer to the camera appear lighter, those who are further appear darker.

Summary

7



1. Stereo camera calibration → compute camera relative pose
   - Epipolar rectification → align images
2. Search correspondences
3. Output: compute stereo triangulation or disparity map
4. Consider baseline and image resolution to compute accuracy!

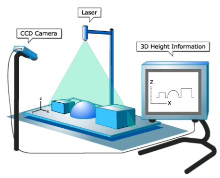
Range sensors

8

- Sonar

- Laser range finder

- Time of Flight Camera
- Structured light



Range: Time-of-Flight

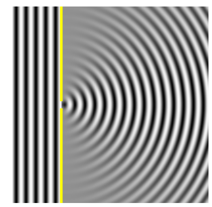
4.1.6

9

- Time-of-flight uses *propagation speed* of waves
  - Sound or electromagnetic
- Distance traveled by a wave is:

$$d = c \cdot t$$

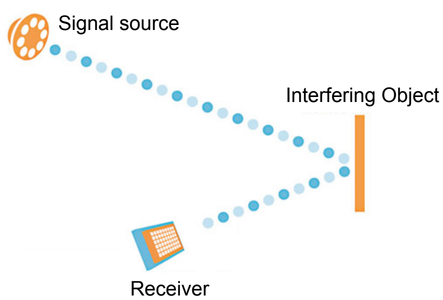
d = distance traveled (round-trip)
c = speed of wave propagation
t = time of flight



Time-of-Flight

4.1.6

10



Time-of-Flight: Accuracy

11

- Sources of inaccuracy:
  - Uncertainties about exact time of arrival of the reflected signal
  - Inaccuracies in the time of flight measure (laser range sensors)
  - Opening angle of transmitted beam (ultrasonic range sensors)
  - Interaction with the target (surface, specular reflections)
- Variation of propagation speed
  - Propagation speed of sound: 0.3 m/ms
  - Propagation speed of of electromagnetic signals: 0.3 m/ns
    - One million times faster.
    - Laser range sensors expensive and delicate.
- Speed of mobile robot and target

Scanning Range Sensing

12

- Confidence in the range (phase estimate) is inversely proportional to the square of the received signal amplitude.
- Dark or distant objects → worse estimates than closer brighter objects

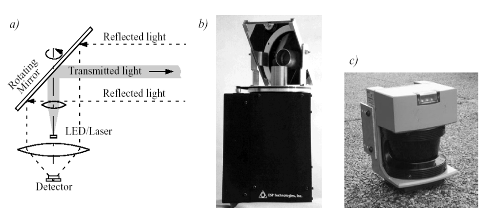
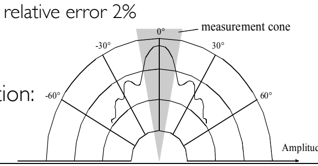


Figure 4.11
(a) Schematic drawing of laser range sensor with rotating mirror; (b) Scanning range sensor from EPS Technologies Inc.; (c) Industrial 180 degree laser range sensor from Sick Inc., Germany

Sonar

13

- Typical frequency: 40kHz - 180 kHz
  - Lower frequencies correspond to longer range
  - Sound from piezo transducer
  - Transmitter and receiver separated or not separated
- Range between 12 cm up to 5 m
  - (Ideal) resolution of ~2 cm
  - (Ideal) Accuracy 98% => relative error 2%
- ~Conical propagation
- Typical intensity distribution:



Sonar: Speed

14

- Transmit a packet of (ultrasonic) pressure waves
- Distance d of the echoing object can be found from propagation speed of sound c and the time of flight t. $d = \frac{c \cdot t}{2}$
- Speed of sound c (340 m/s) in air is: $c = \sqrt{\gamma \cdot R \cdot T}$
- Where
  - γ: adiabatic index (isentropic expansion factor)
  - R: gas constant
  - T: temperature in degree Kelvin

Sonar: Bandwidth

15

- An object that is 3 m away will take 20 ms, limiting its operating speed to 50 Hz.
- This update rate can affect maximum speed possible while still sensing and avoiding obstacles safely.
- But if the robot has a ring of 20 ultrasonic sensors, each firing sequentially and measuring to minimize interference between the sensors, then the ring's cycle time becomes 0.4 seconds => frequency of each one sensor = 2.5 Hz.

Laser Range Sensor

17

- Slightly deprecated term: LIDAR
- Similar to sonar
  - Without the signal speed issues
- More accessile, robust and cheaper than ever before

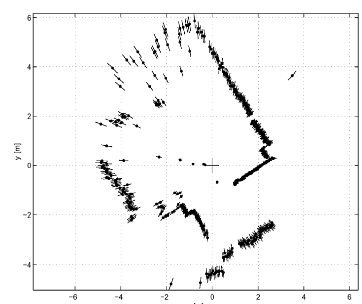

SICK


Hokuyo

Example of Scanning

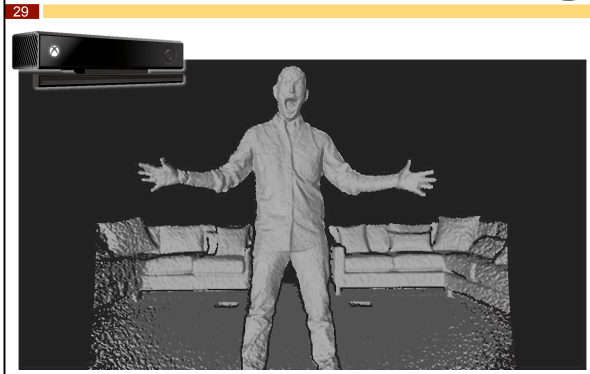
23

- Length of the lines through measurement points indicate uncertainty



Modern Time-of-Flight

29

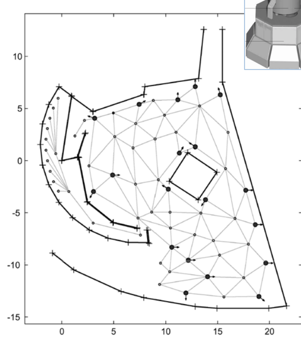


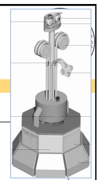
Line and Plane Features

Environment Mapping

32

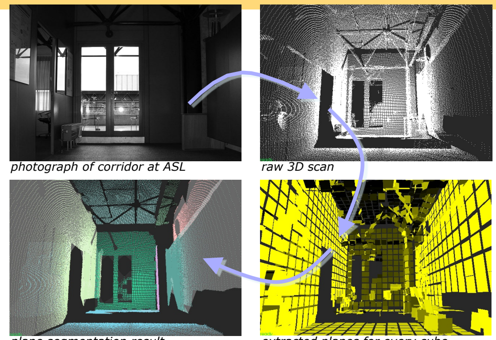
- Features for Localization
  - Compact map 26 bytes / m2
  - Multi-hypothesis tracking
- Topological map for global planning
- Raw data for local planning and obstacle avoidance





Example Result

35

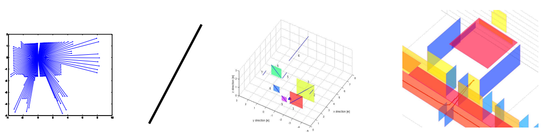


photograph of corridor at ASL
raw 3D scan
plane segmentation result
extracted planes for every cube

Features: Motivation

36

- Why Features?
  - Raw data: huge amount of data to be stored*
  - Compact features require less storage (e.g. Lines, planes)
  - Provides rich and accurate information
  - Basis for high level features (e.g. more abstract features, objects)



Line Extraction

37

- Algorithms:
  - Split and merge
  - Linear regression
  - RANSAC
  - Hough-Transform

Line Extraction: Motivation

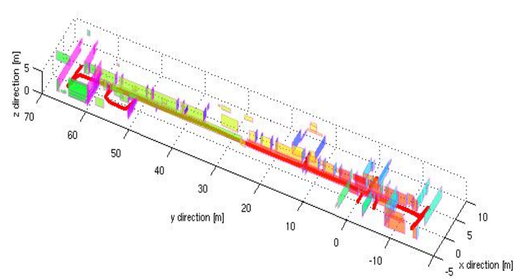
38

-

Line Extraction: Motivation

39

- Map of the ASL hallway built using orthogonal planes constructed from line segments



Line Extraction: Motivation

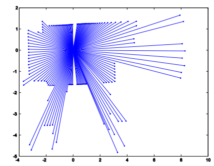
40

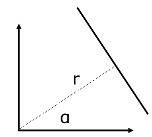
- Why laser scanner:
  - Dense and accurate range measurements
  - High sampling rate, high angular resolution
  - Good range distance and resolution.
- Why line segment:
  - The simplest geometric primitive
  - Compact, requires less storage
  - Provides rich and accurate information
  - Represents most office-like environment.

Line Extraction: The Problem

41

- Scan point in polar form: (ρ_i, θ_i)
- Assumptions:
  - Gaussian noise with $(0, \sigma)$ for ρ
  - Negligible angular uncertainty
- Line model in polar form:
  - $x \cos \alpha + y \sin \alpha = r$
  - $-\pi < \alpha <= \pi$
  - $r >= 0$





Line Extraction: The Problem

42

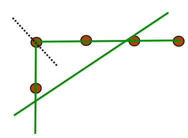
- Three main problems:

1. How many lines?
2. Which points belong to which line?
   - This problem is called SEGMENTATION
3. Given points that belong to a line, how to estimate parameters?
   - This problem is called LINE FITTING

Split-and-Merge

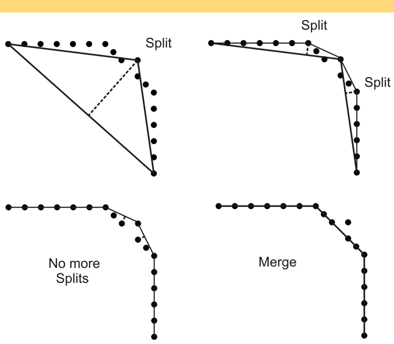
43

- The most popular algorithm
- Originated from computer vision.
- A recursive procedure of fitting and splitting.
- A slightly different version, called Iterative-End-Point-Fit, simply connects the end points for line fitting.



Split-and-Merge

44



Algorithm 1: Split-and-Merge

45

Algorithm 1: *Split-and-Merge*

1. Initial: set s_1 consists of N points. Put s_1 in a list L
2. Fit a line to the next set s_i in L
3. Detect point P with maximum distance d_P to the line
4. If d_P is less than a threshold, continue (go to step 2)
5. Otherwise, split s_i at P into s_{i1} and s_{i2}, replace s_i in L by s_{i1} and s_{i2}, continue (go to 2)
6. When all sets (segments) in L have been checked, merge collinear segments.

Split-and-Merge Example

46