Mobile Robot Systems

Lecture 4: Sensors & Perception

Dr. Amanda Prorok

asp45@cam.ac.uk

www.proroklab.org

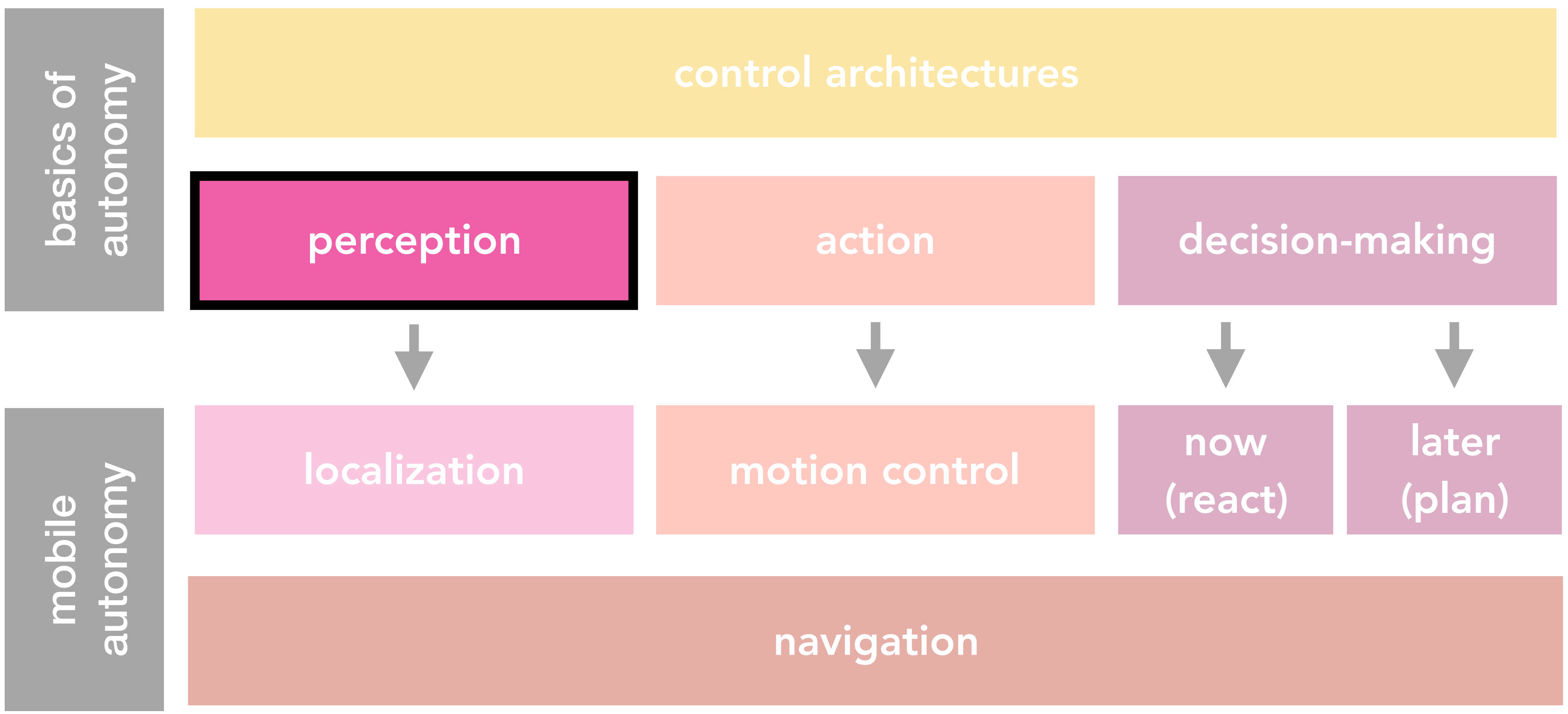
UNIVERSITY OF CAMBRIDGE

In this Lecture

- Introduction to sensing and perception for autonomous robots
- Introduction to a few popular sensors
  - Basic HW concept
  - Sensor model
  - Application
- Odometry

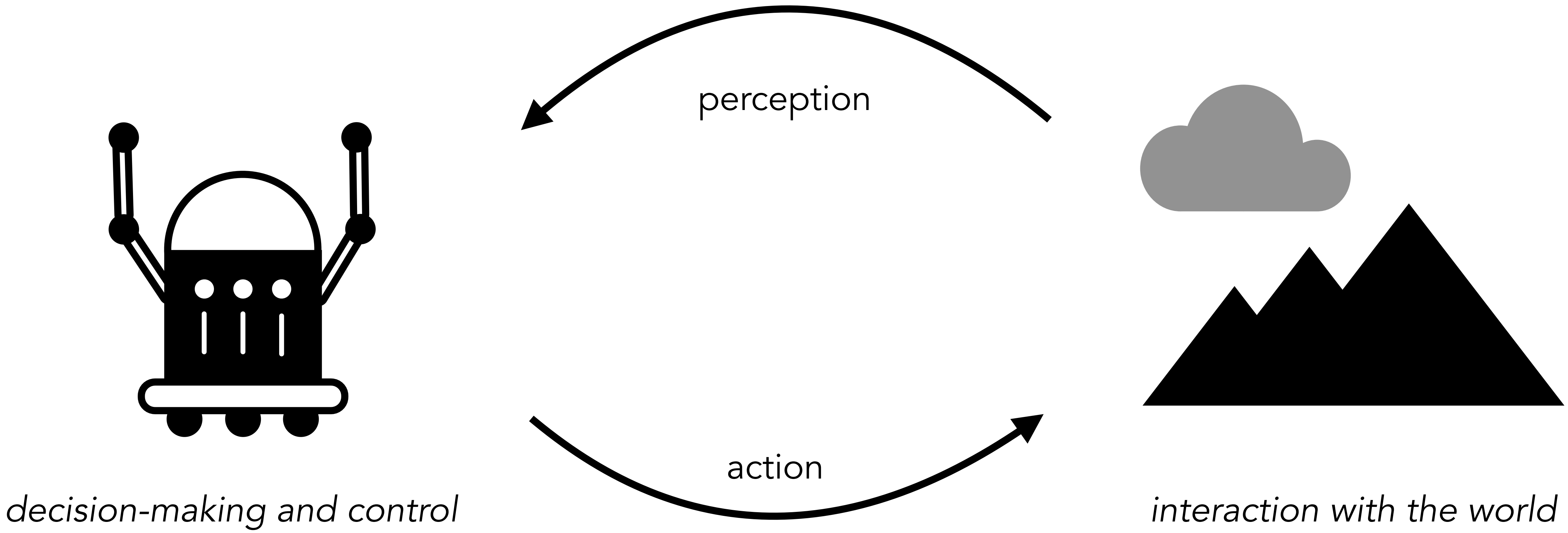
- Credits:
  - Odometry example for Thrun's book (Probabilistic Robotics)
  - MLE example from Zisserman's course (Oxford)

Control Architectures



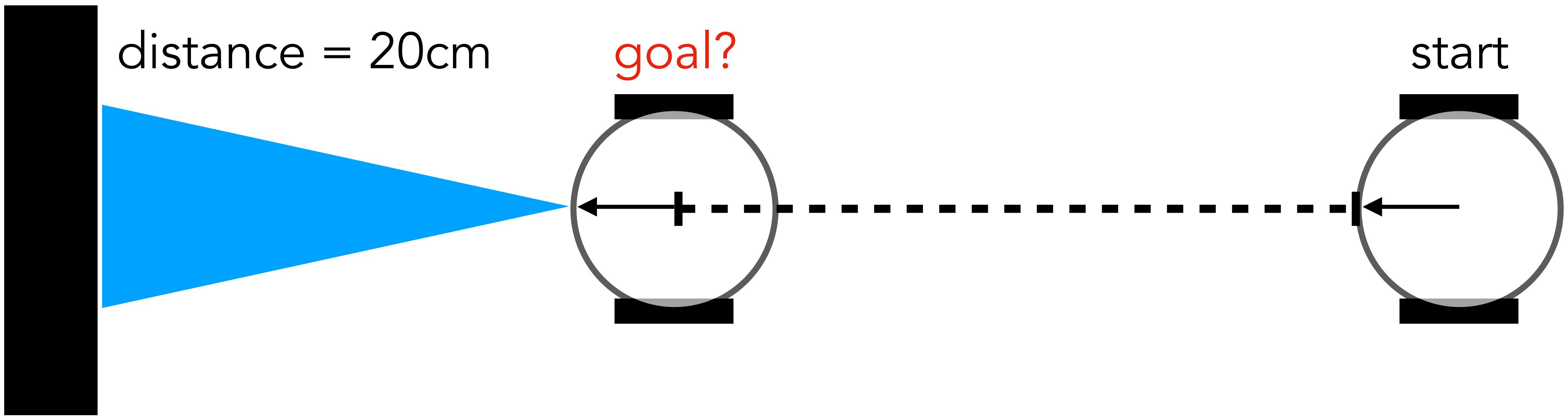
Perception-Action Loop

- Basic building block of autonomy



Perceiving the Environment

- A sensor is a component that measure some aspects about the state of the world or the state of the robot.
- Recall open-loop vs. closed-loop:
  - Example: a robot is to move towards a wall and stop 20cm away from it. Floor unevenness, friction, and other environmental factors mean that the robot cannot execute open-loop control to arrive at the target pose.



Classification of Sensors

measure something **external** to the robot

measure something **internal** to the robot

exteroceptive

proprioceptive

active

- infrared (distance)
- sonar range finder
- laser scanners (distance)
- …

affect environment by emitting energy

passive

- ambient light
- sound (microphone)
- camera (vision)
- …

do not affect environment

- odometry
- speedometer
- energy level
- accelerometer
- …

Distance Sensors

- Most distance sensors are active: emit a signal and receive its reflection from an object (if any).
- Two principles:
  - (round-trip) time-of-flight (TOF); the signal travel speed is known

$$s = \frac{1}{2}vt$$

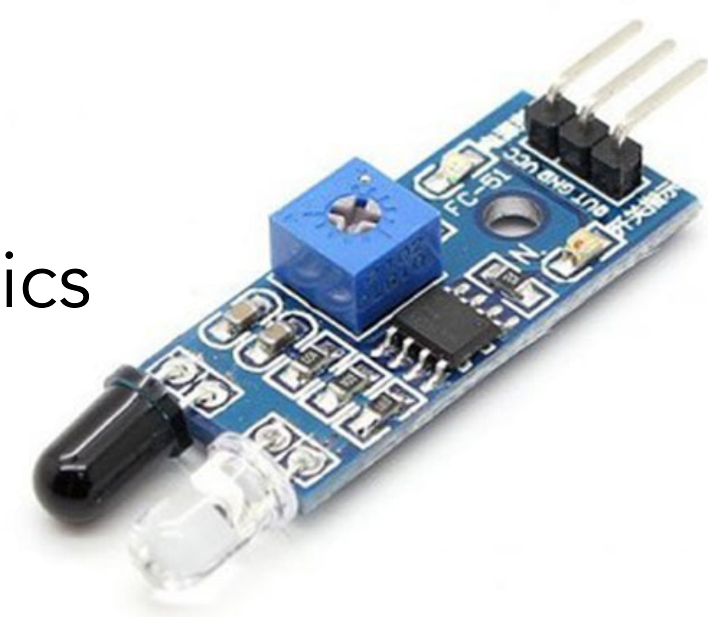
  - received signal intensity; the signal attenuation is known

Example signal modalities:

1. Infrared
2. Ultrasound
3. Coherent light (laser)

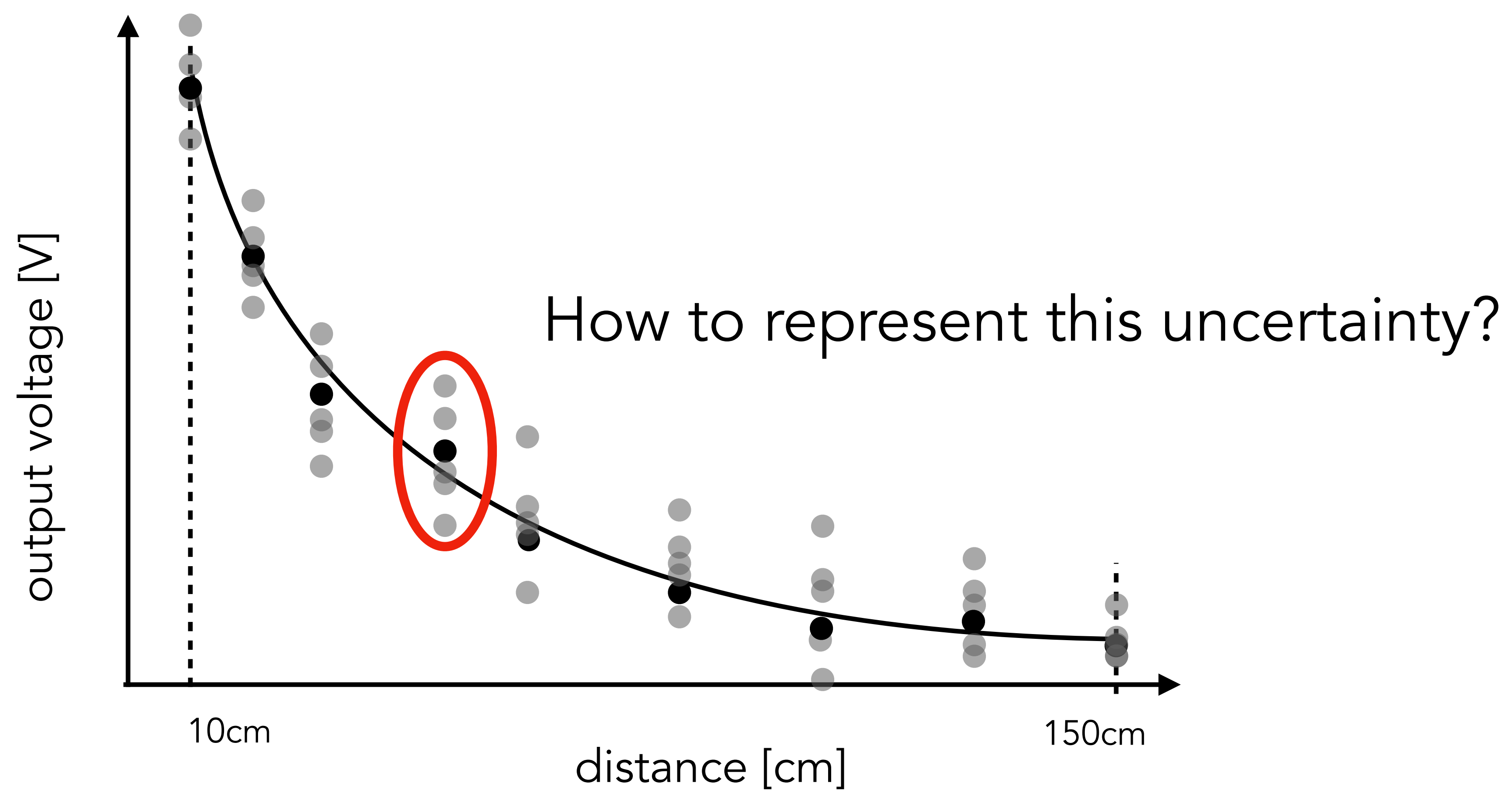
Infrared Proximity Sensors

- Infrared light has long wavelengths (700-1000nm); invisible to human eye
- Reflectance sensing hardware: The emitter is usually made with a light-emitting diode (LED), and the detector is usually a photodiode/phototransistor.
- Measurement principle: Detect the presence of an object by measuring the intensity of the reflected light
- When ambient light must be subtracted, sensor takes a measurement without emission.
- Common usage: cheap robots (e.g., for education)
- Main disadvantage: depends on object characteristics (shape, color, surface)



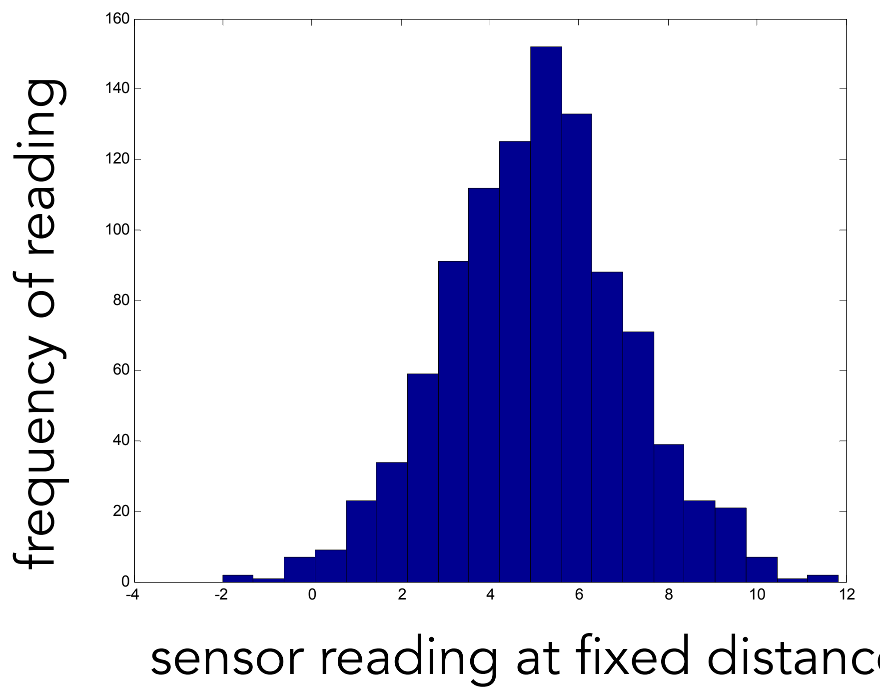
Sensor Calibration

- How to relate a sensor measurement to a perceptive feature?
- Example: calibrate an infrared proximity sensor
- Aim: determine mapping between sensor reading and desired feature
  - Step 1: use ground-truth telemetry system to set up sensor
  - Step 2: measure and tabulate values
  - Step 3: Fit the curve.



Representing Uncertainty

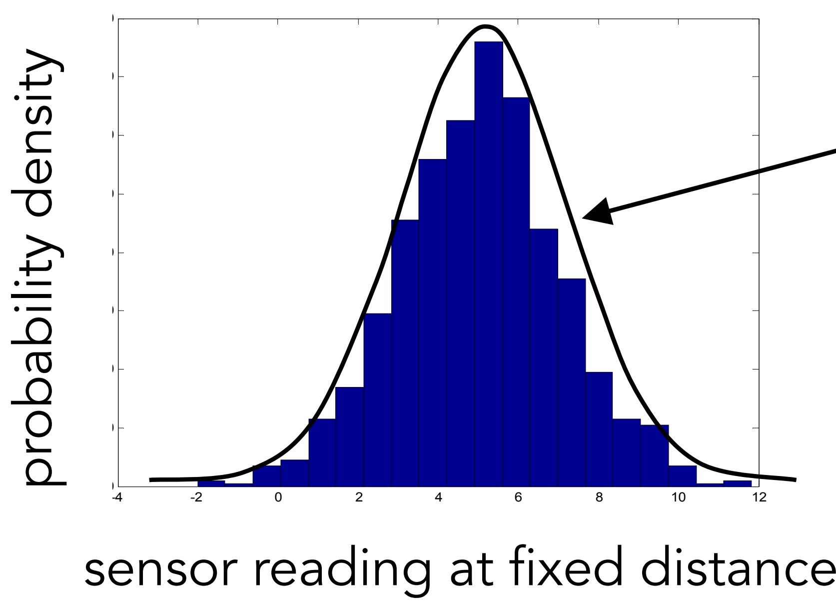
- Sensors are imperfect devices
  - Random errors as well as systematic errors.
  - **What is the uncertainty in a given sensor measurement?**
- We need a representation for random errors associated to a given sensor.
- Repeat measurements and describe the sensor's distribution
- Options:
  1. store original measurements (x_i, z_i) — *true* x_i, *measured* z_i
  2. store histogram of measurements p_i
  3. compute a compact representation of distribution



Representing Uncertainty

- Compact representation that describes z: fit a model through moment matching.
- Assumption: values are normally distributed. A Gaussian distribution can be described through 1st and 2nd moments.

mean: $\mathbb{E}[z] = \sum_{i=1}^{n} p_i z_i = \mu$ variance: $Var(z) = \sum_{i=1}^{n} p_i (z_i - \mu)^2$

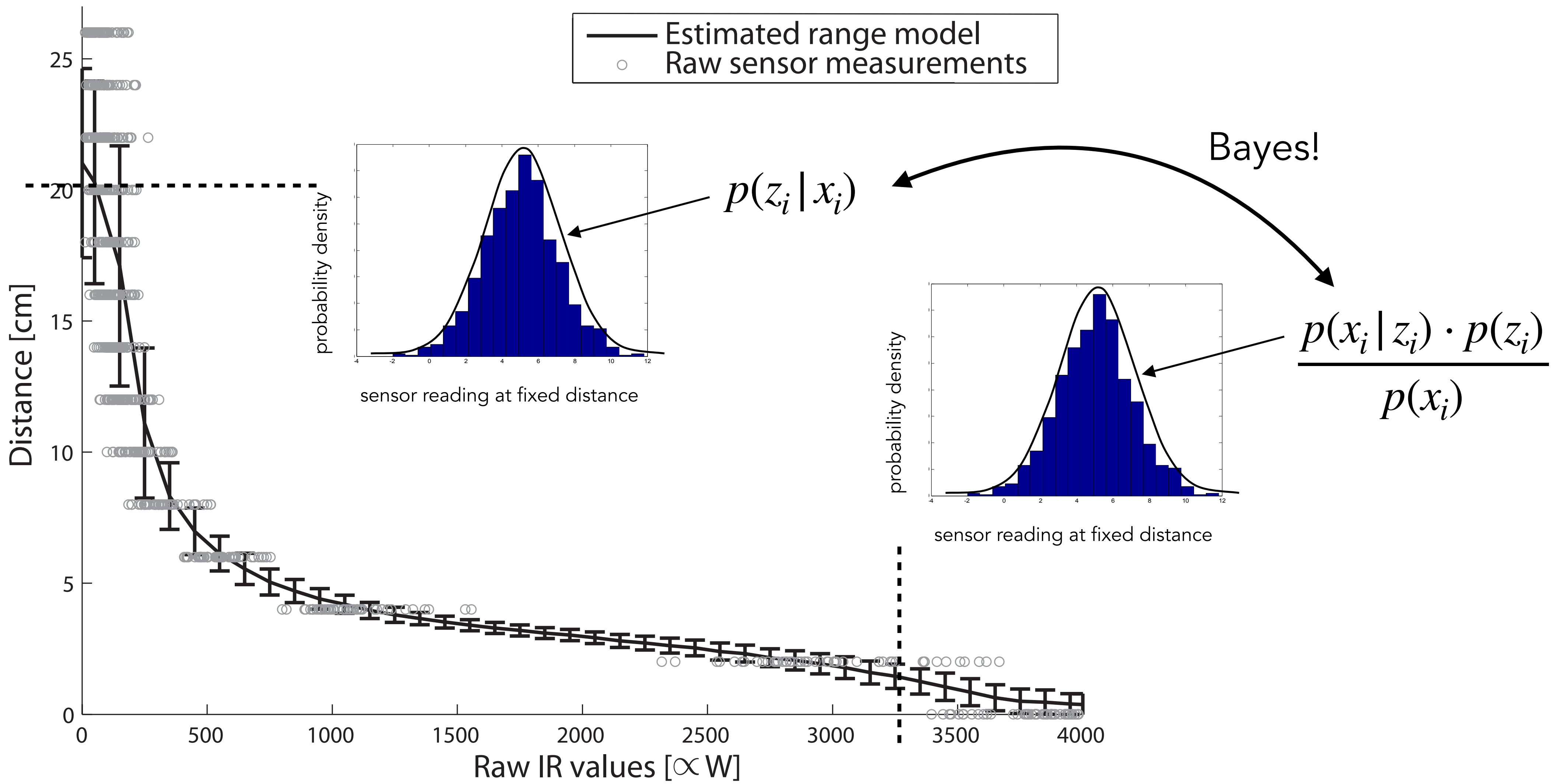


$p(z_i | x_i)$

fitted model for measurement z_i

IR Sensor Model

- What is the uncertainty around a given sensor measurement?
- Collect data for all distances in operational range.



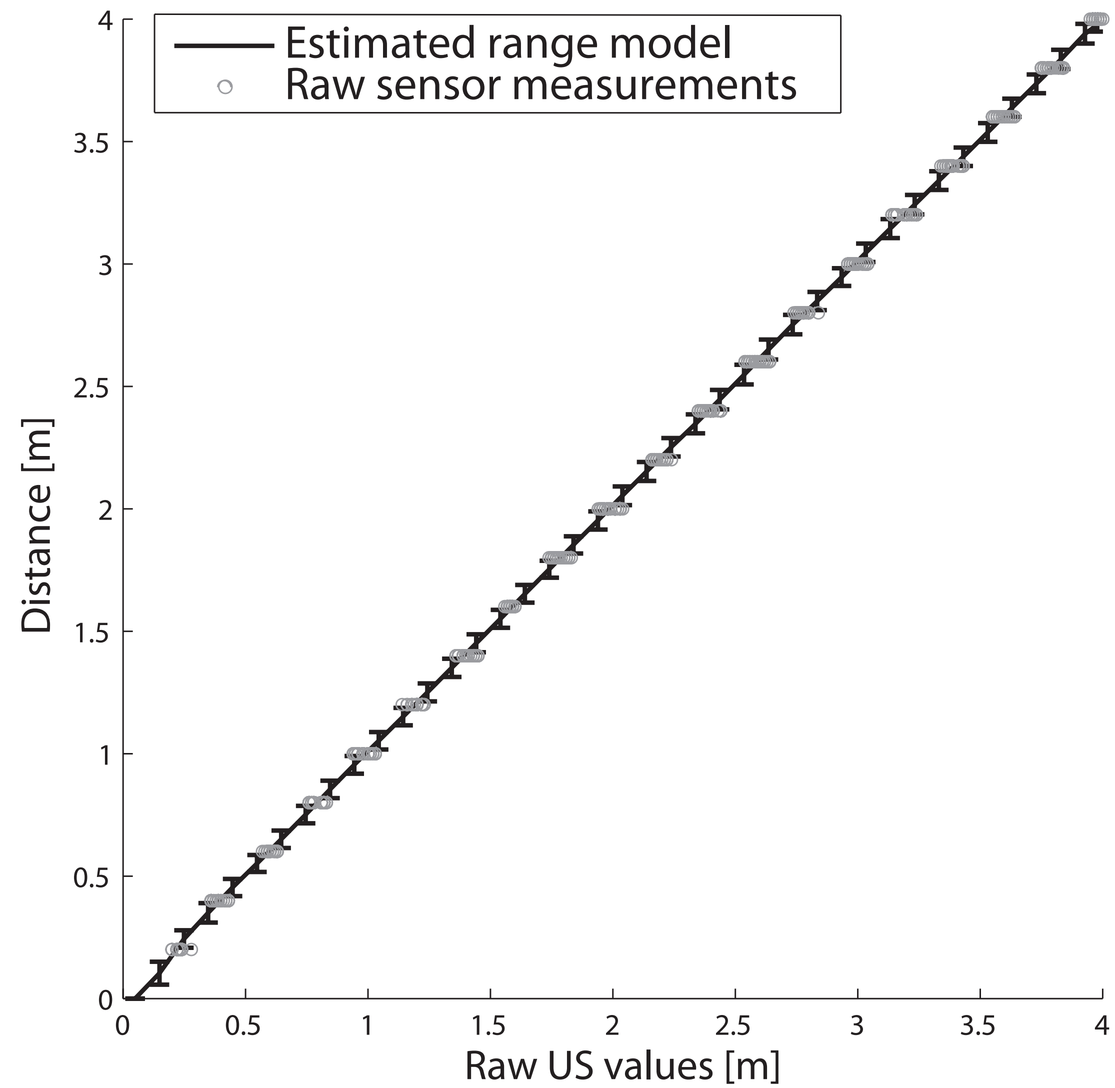
A. Prorok et al., Indoor Navigation Research with the KIII Mobile Robot: An Experimental Baseline with a Case-Study on UWB Positioning, 2010

Ultrasonic (Sonar) Sensors

- Emitter produces a chirp or ping (of ultrasound frequency > 20kHz; inaudible to human ears)
- A timer is started when the chirp is emitted. If the wave encounters a barrier, it is reflected and measured upon return.
- Hardware: a transducer (transforms one form of energy to another), where mechanical energy is transformed to sound as the membrane of the sensor flexes; a microphone (detector)
- Measurement principle: RTOF, with speed of sound: at room temperature it is 343m/s
- Common usage: underwater applications (sound travels well in water)
- Main disadvantage: specular reflections (e.g., try measuring distance to object at steep angle); sensor cross-talk (interference)

US Sensor Model

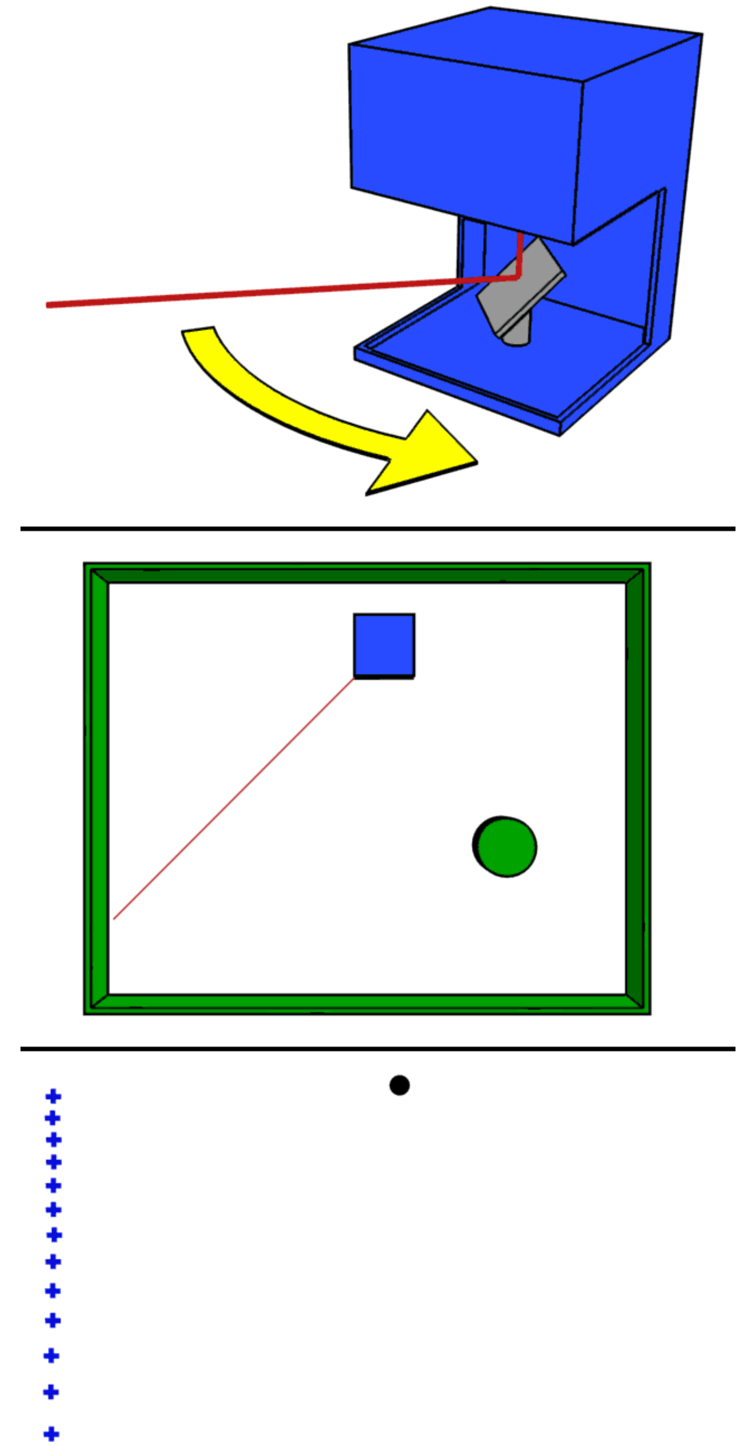
- Collect data for all distances in operational range for reflection off of a planar surface.



A. Prorok et al., Indoor Navigation Research with the KIII Mobile Robot: An Experimental Baseline with a Case-Study on UWB Positioning, 2010

Laser Scanner (LIDAR)

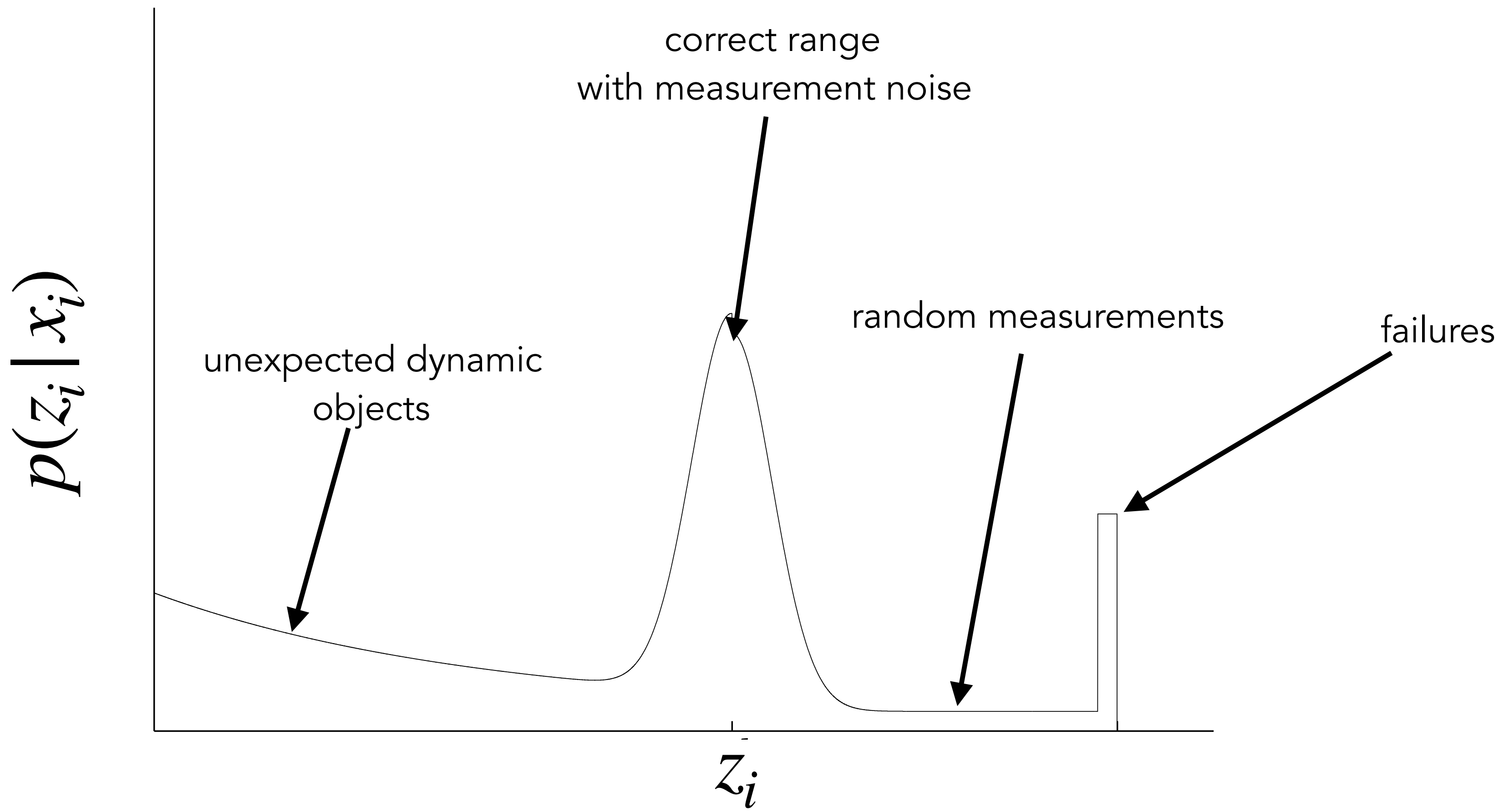
- LIDAR: **li**ght **d**etection **a**nd **r**anging
- Measure the TOF of a pulse of light.
- Coherent light (usually infrared). All the energy is concentrated in a narrow beam.
- Distance can be computed by measuring the time of flight; e.g., for 30cm round-trip TOF is 0.002μs
- Hardware: receiver/emitter pairs (channels); combined with rotating mirrors for sweeping.
- Common usage: high-end research robots; autonomous vehicles
- Advantages: very high sampling rates possible; no interference between emitted beams; precision
- Disadvantages: moving parts (high energy usage); expensive, large, heavy; affected by weather



* By Mike1024, via Wikimedia Commons

LIDAR Sensor Model

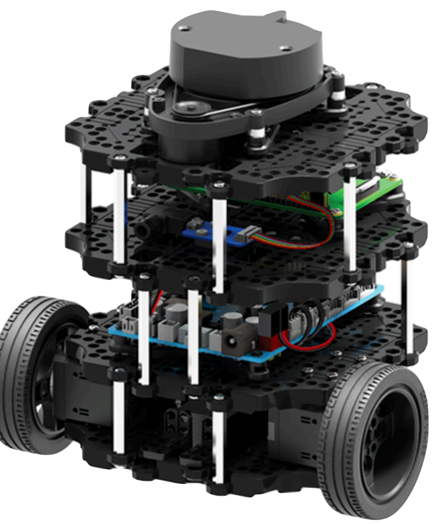
- Range finders have may have multiple components to their uncertainty models. The model depends on the environment, this is an example:



- The likelihood of data Z with associated positions X for a given map (environment) M is given by: $p(Z \mid X, M, \Theta)$ which can be determined through Maximum Likelihood estimators.

*image: Probabilistic Robotics; Thrun et al.

Laser Scanner (LIDAR)

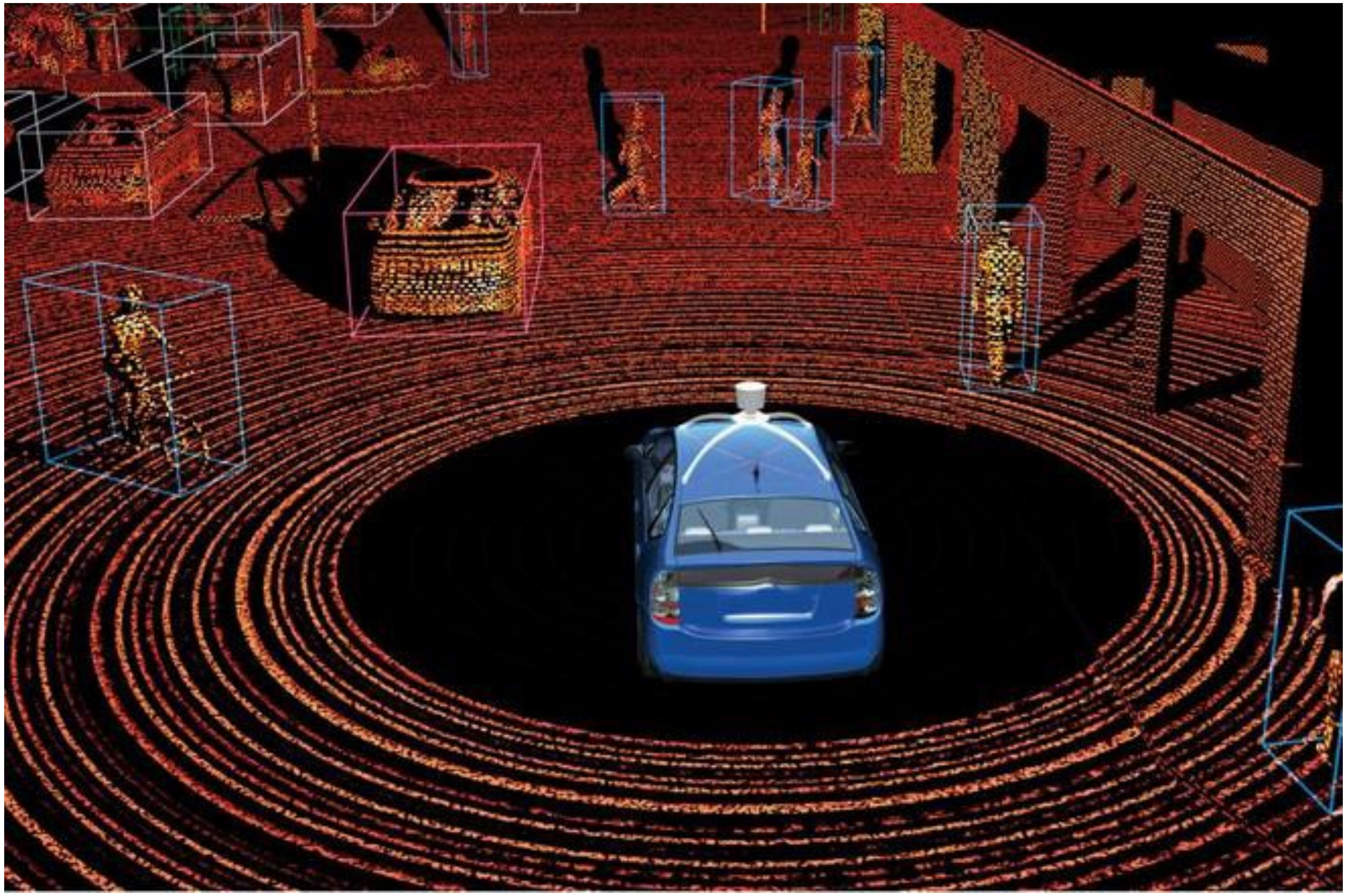


- Turtlebot III Range Finder:
  - scanning area: 360°
  - resolution: 1°
  - detection range: 0.12 - 3.5m
  - price: < USD 500

- Hokuyo UTM-30LX
  - scanning area: 240°
  - resolution: 0.25°
  - detection range: 0.002 - 30m
  - price: > USD 5000

- Velodyne VLS-128
  - scanning area: 360°
  - resolution: 0.11°; up to 9.6 mio. points per second
  - detection range: up to 300m
  - price: > USD 24’000

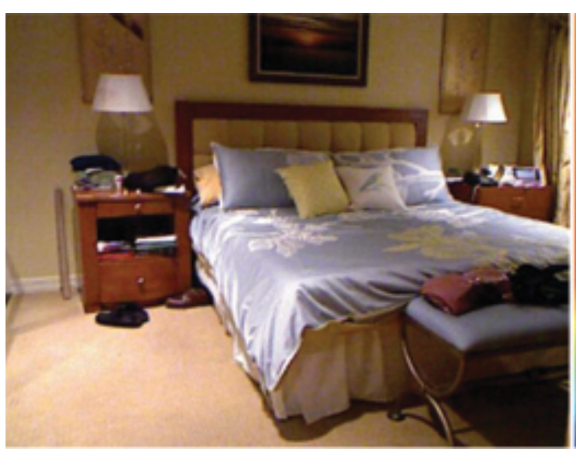
Laser Scanner (LIDAR)



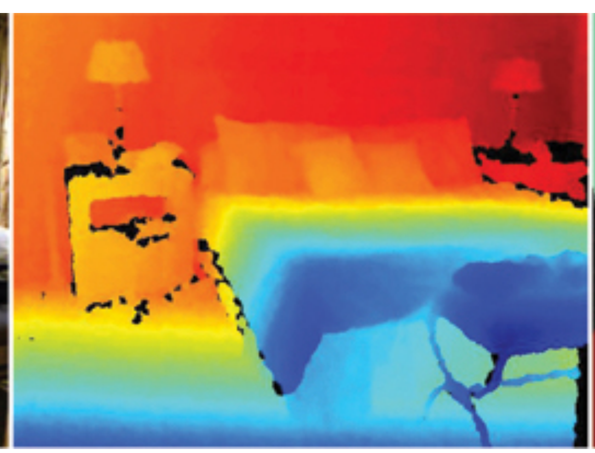
- Point clouds are collected (e.g., 1 mio. data points per second)
- Disadvantages: price; energy consumption; size; does not work well in bad weather (e.g., rain)

Vision

- Vision for robotics is concerned with different problems than classical computer vision:
  - must guarantee real-time, fast operation
  - perception is task-driven (less general)
  - vision in motion: streams (rather than still images)
  - applications: object tracking, visual-odometry, information gathering

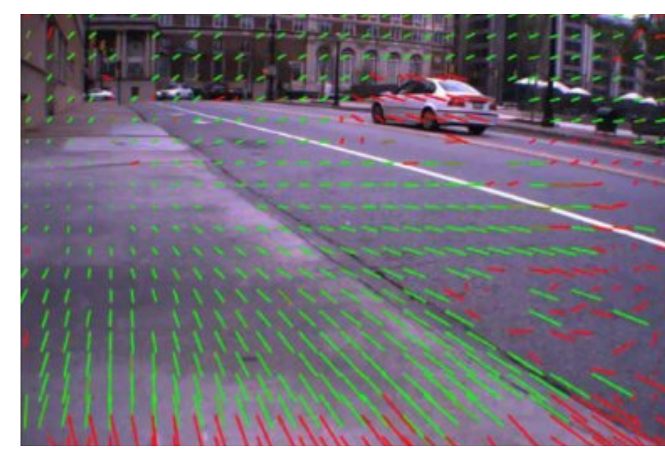


RGB image



raw depth

segmentation



optical flow

* image credit: N. Silberman NYU

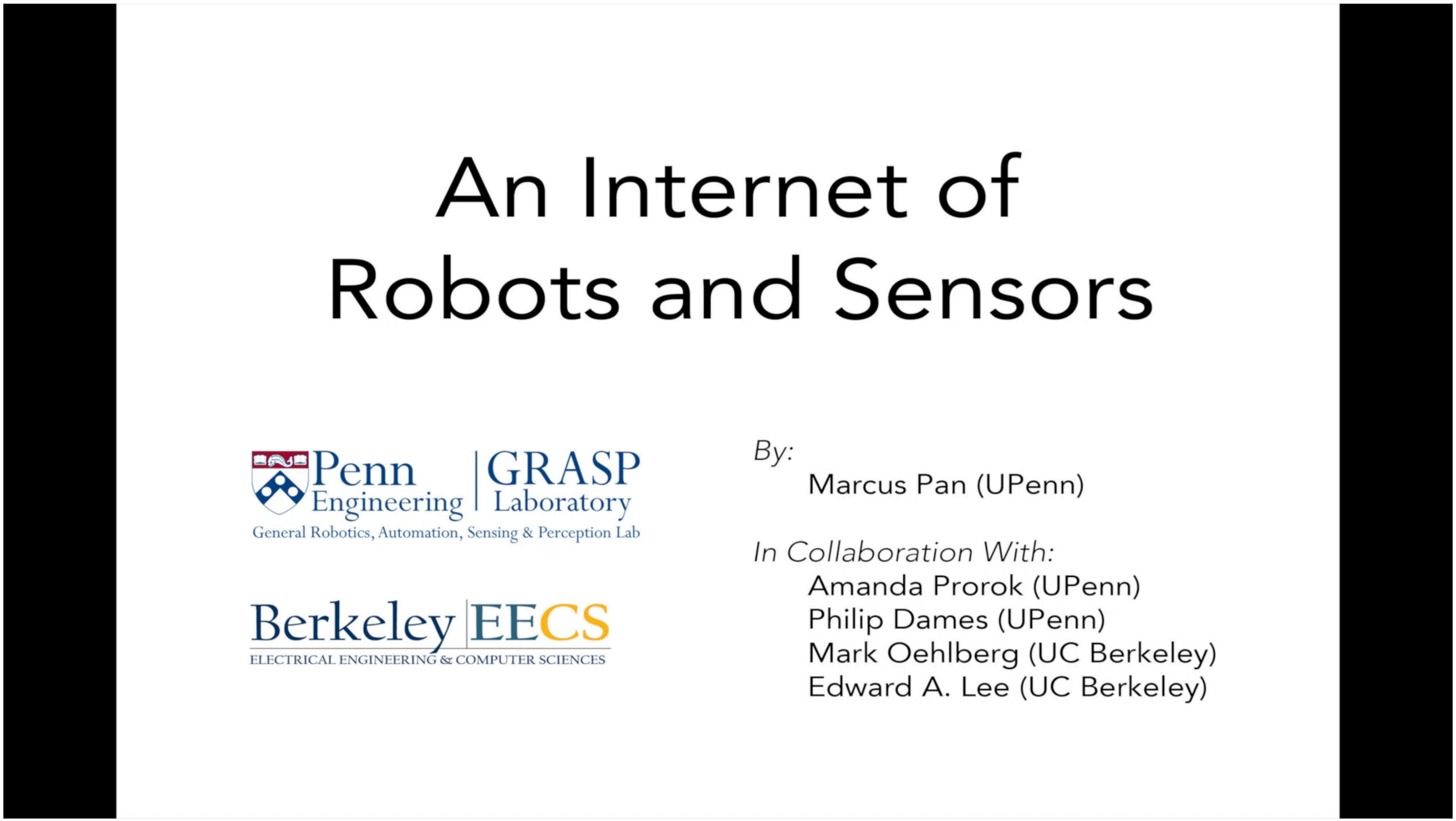
Fiducials

- **Fiducial markers** (AprilTag, U. Michigan, E. Olson et al.)
  - Place easy to recognize landmarks in your environment
  - AprilTags are conceptually similar to QR Codes (bar code) and use quad detection and a smart coding scheme.
  - A tag encodes between 4 and 12 bits
  - Detection of identity
  - Designed for high localization accuracy (3D pose w.r.t. camera): 6 DOF localization of features from a single image.
  - Reference paper: AprilTag: A robust and flexible visual fiducial system; Edwin Olson; ICRA, 2011

* https://april.eecs.umich.edu/software/apriltag

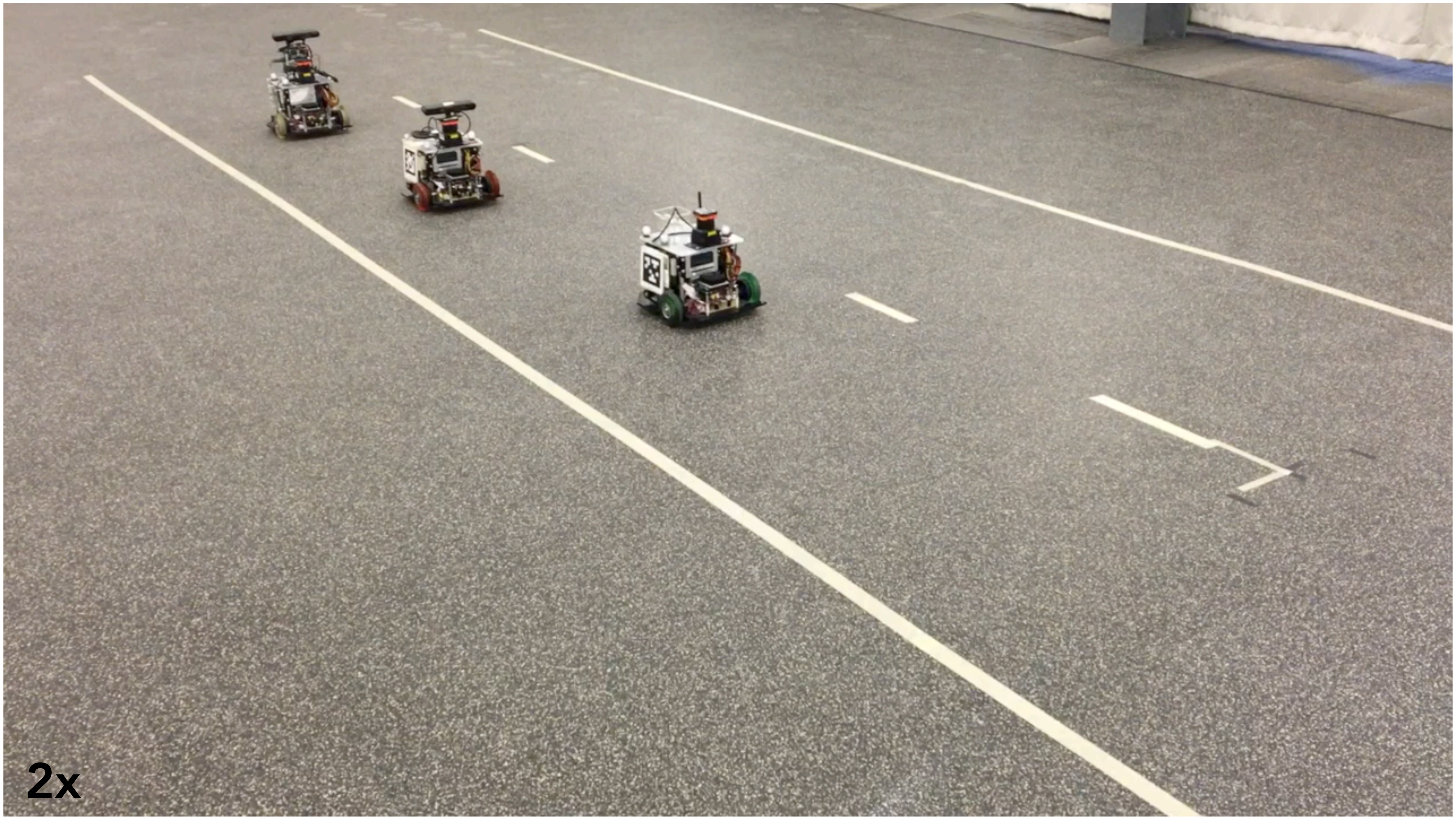
Mobile Robot & IoT

- Using fiducial markers to create a robotic-IoT setup.



Leader-Follower Setup

- Using fiducial markers to find distance and bearing to leader robot.



Whitzer, Kennedy, Prorok, Kumar; 2016

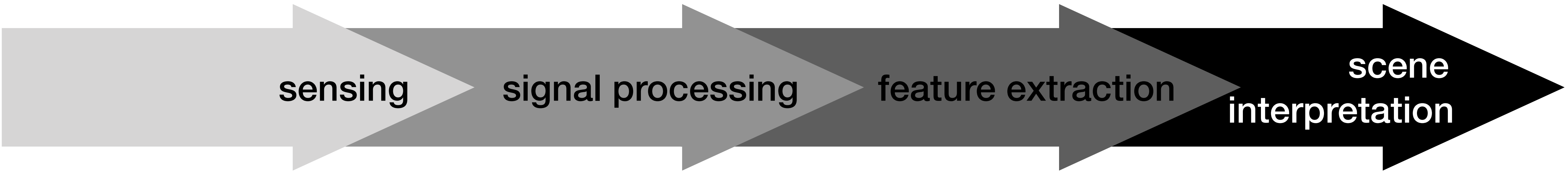
Considerations

- Sensor response rating:
  - Dynamic range: spread between lower and upper limits of input values; often measured as a ratio of max/min input.
  - Resolution: min. distance between two values that can be detected by a sensor.
  - Linearity: behavior of output as input varies.
  - Frequency: speed at which sensor can provide readings.
- Sensor performance:
  - Sensitivity (and cross-sensitivity): degree to which incremental change in target input signal changes output signal; cross-sensitivity is undesirable
  - Errors: accuracy and precision; random vs systematic error;

accuracy: $1 - \frac{|error|}{true}$ *precision:* $\frac{range}{\sigma}$

- Other factors: price, energy consumption, size, weight…

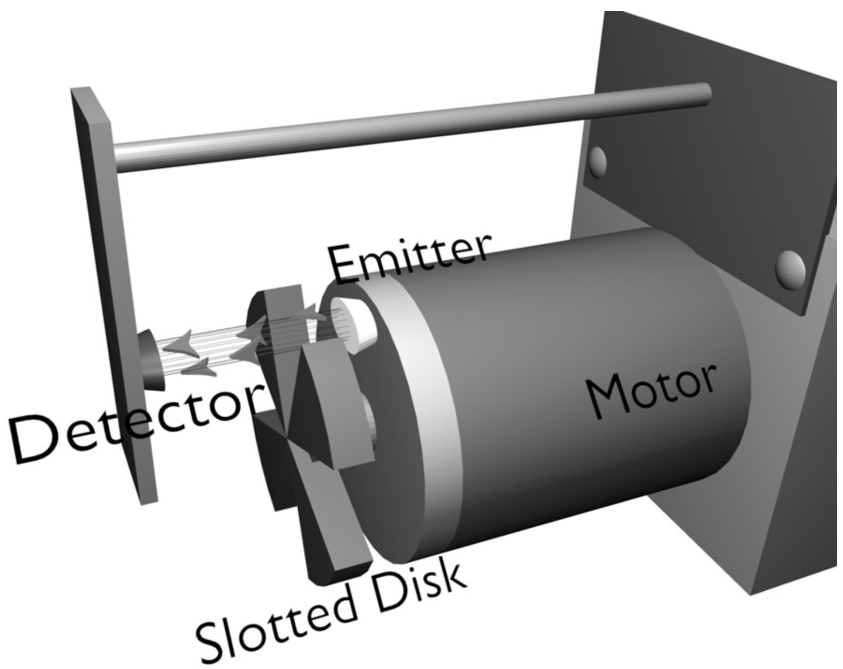
Perceptual Pipeline



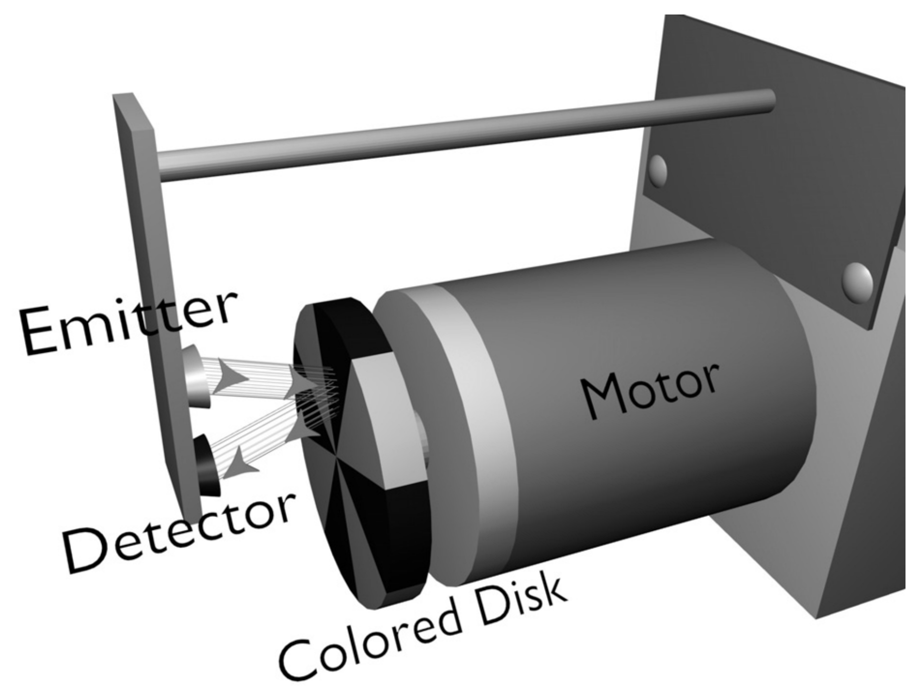
- How to control robot behavior?
  - **Proximal architecture**: robot behavior is close to sensor readings (with actions being triggered by some low-level model).
  - **Distal architecture**: sensor readings are processed to generate a higher-level model that informs a robot's scene interpretation.
- Feature extraction is the process of generating higher-level perception. This is useful for long-term, more sophisticated tasks.
  - E.g., map building and route planning (see next lectures)

Odometry

- Application of light sensors for speedometers / odometers
- Shaft encoding is the underlying sensing mechanism.



break-beam shaft encoder

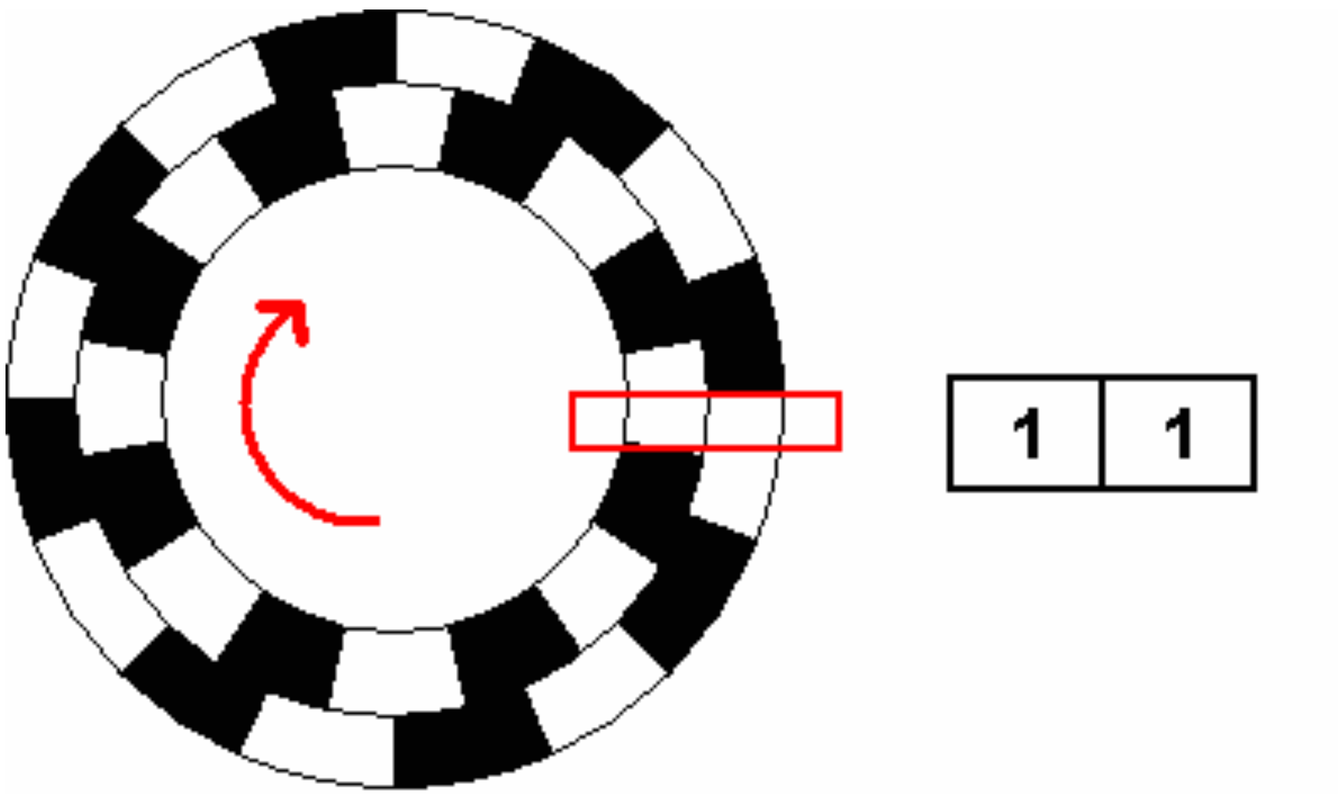


reflectance-based shaft encoder

* image credit: The Robotics Primer (Mataric)

Odometry

- How can we detect the direction of motion?
- Quadrature encoder detects direction of motion. A 2nd detector is shifted by a phase of 90° that allows for determining the sense of rotation.



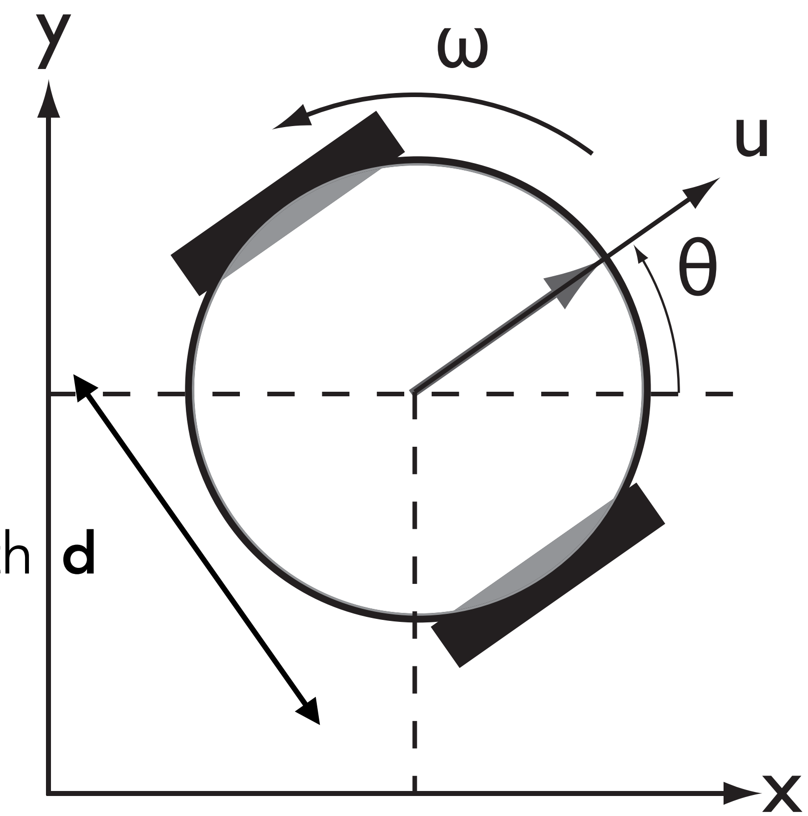
* image credit: Matt Hercules (Wikipedia)

Odometry

- Compute updated position based on left and right wheel readings.

current pose:

$$x_t = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}$$



traveled distance:

$$\Delta s = \frac{\Delta s_r + \Delta s_l}{2}$$

path traveled in last sampling interval:

$$\Delta x = \Delta s \cos(\theta + \Delta\theta/2)$$

$$\Delta y = \Delta s \sin(\theta + \Delta\theta/2)$$

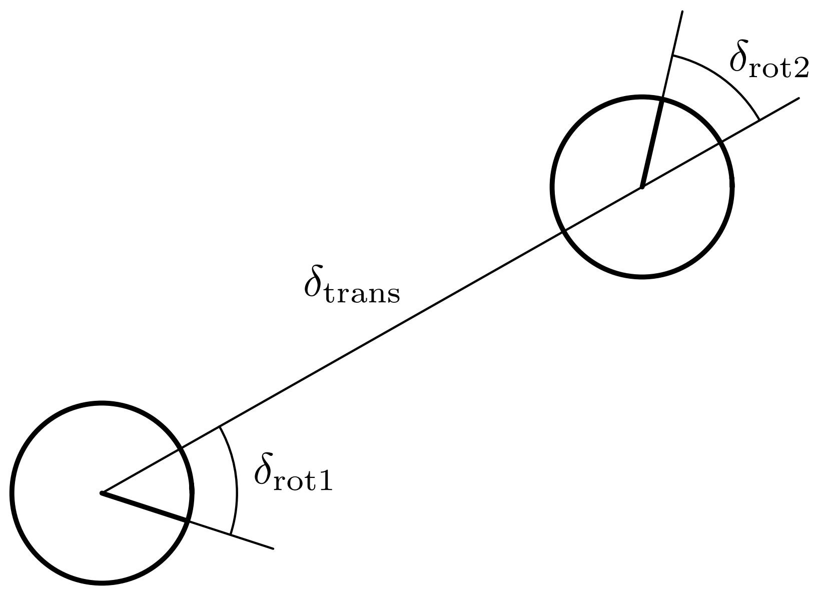
$$\Delta\theta = \frac{\Delta s_r - \Delta s_l}{d}$$

updated pose:

$$x_{t+1} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} \Delta s \cos(\theta + \Delta\theta/2) \\ \Delta s \sin(\theta + \Delta\theta/2) \\ \Delta\theta \end{bmatrix}$$

Odometry Motion Model

- Integrate wheel encoder information over time to obtain odometry.
- Challenge: drift and slippage.
- Key question: **How to model motion uncertainty?**



$x_t = [x, y, \theta]$ robot pose at time t

$\bar{x}_t$ position from odometry measurements

$u_t = (x_{t-1}, \bar{x}_t)$ relative motion information

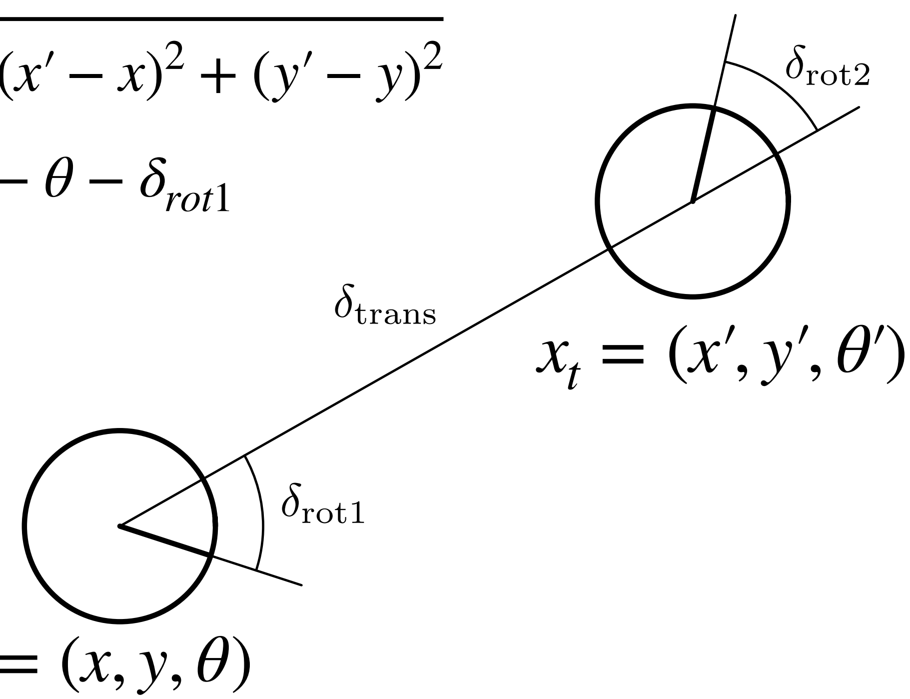
- Odometry model for a differential drive robot: Given a time interval, approximate motion in that interval by two rotations and one translation, all of which are noisy: $(\delta_{\mathrm{rot1}}, \delta_{\mathrm{trans}}, \delta_{\mathrm{rot2}})$

* Probabilistic Robotics; Thrun et al.

Odometry Motion Model

- Idea: use error distributions defined over rotational and translational movement to evaluate likelihood of a given robot pose.

$$\delta_{rot1} = \text{arctan2}(y' - y, x' - x)$$
$$\delta_{trans} = \sqrt{(x' - x)^2 + (y' - y)^2}$$
$$\delta_{rot2} = \theta' - \theta - \delta_{rot1}$$

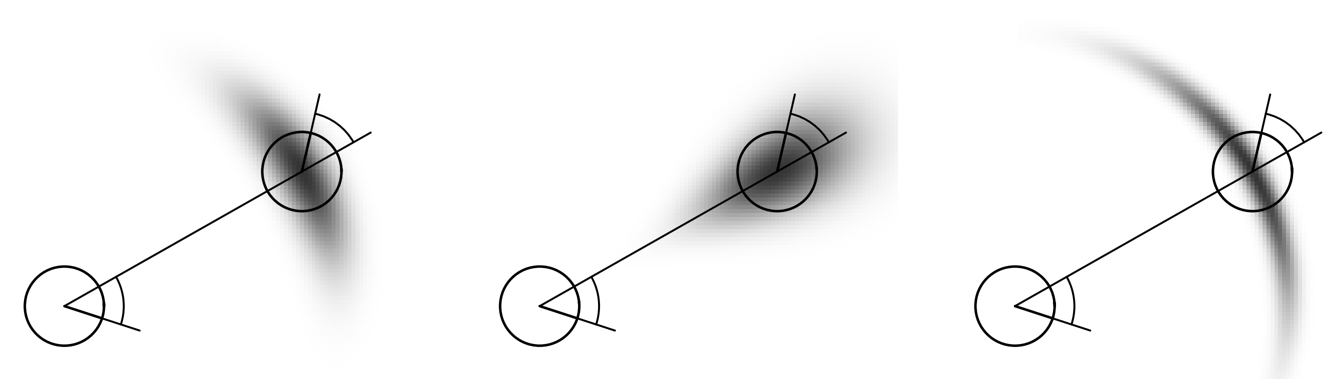


Error distribution over a with mean 0 and variance b:
$\mathbf{prob}(a, b)$

We assume independent error sources:

$$p_1 = \mathbf{prob}(\delta_{\mathrm{rot1}} - \hat{\delta}_{\mathrm{rot1}}, \alpha_1 \hat{\delta}_{\mathrm{rot1}} + \alpha_2 \hat{\delta}_{\mathrm{trans}})$$
$$p_2 = \mathbf{prob}(\delta_{\mathrm{trans}} - \hat{\delta}_{\mathrm{trans}}, \alpha_3\ \hat{\delta}_{\mathrm{trans}} + \alpha_4 (\hat{\delta}_{\mathrm{rot1}} + \hat{\delta}_{\mathrm{rot2}}))$$
$$p_3 = \mathbf{prob}(\delta_{\mathrm{rot2}} - \hat{\delta}_{\mathrm{rot2}}, \alpha_1 \hat{\delta}_{\mathrm{rot2}} + \alpha_2 \hat{\delta}_{\mathrm{trans}})$$

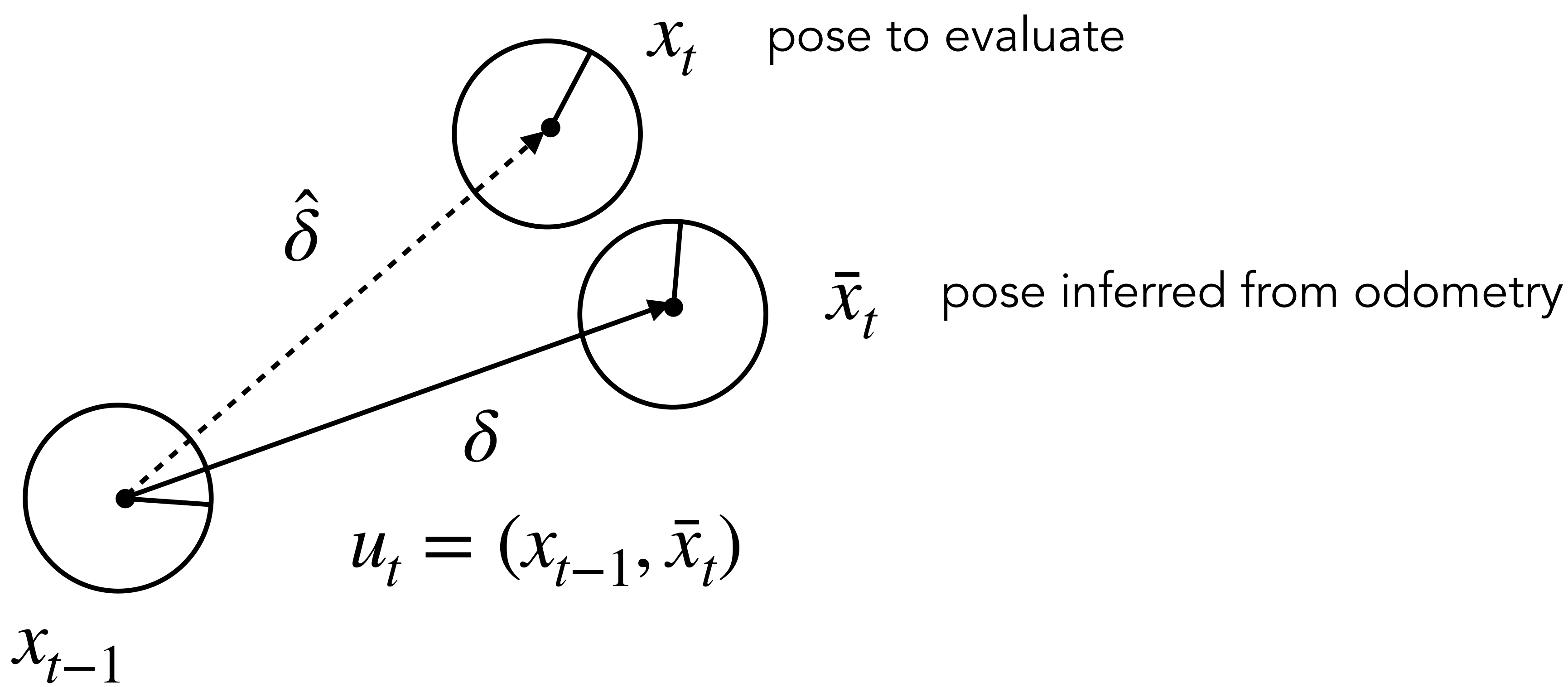
$\alpha_1, \alpha_2, \alpha_3, \alpha_4$ are robot-specific parameters that characterize the noise in robot motion:



* Probabilistic Robotics; Thrun et al.

Odometry Motion Model

- Algorithm to compute likelihood of a pose for given odometry information: $p(x_t \mid u_t, x_{t-1})$



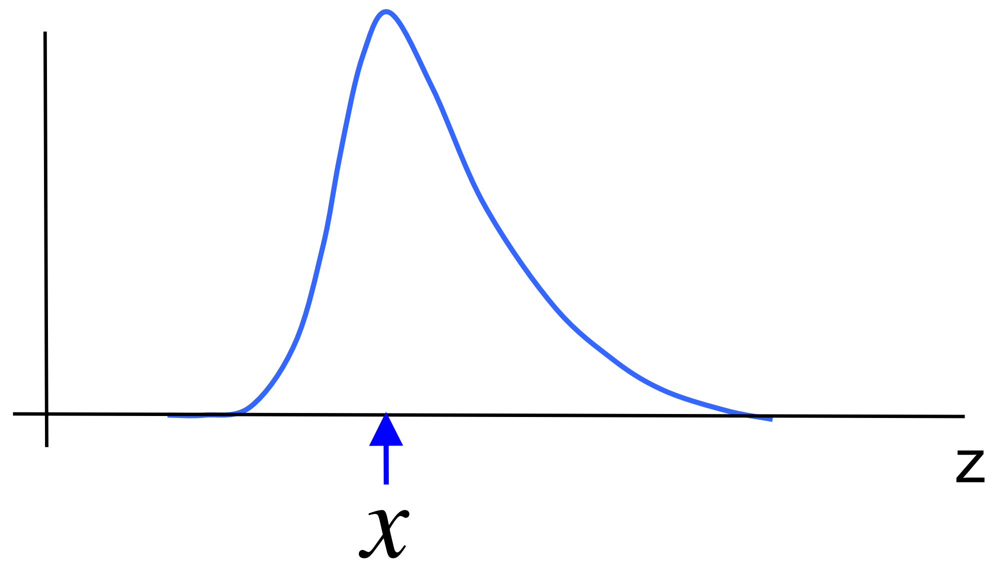
1. Compute $\delta_{rot1}, \delta_{rot2}, \delta_{trans}$ from odometry information $u_t = (x_{t-1}, \bar{x}_t)$
2. Compute $\hat{\delta}_{rot1}, \hat{\delta}_{rot2}, \hat{\delta}_{trans}$ from (x_{t-1}, x_t)
3. Return: $p_1 \cdot p_2 \cdot p_3$

How is a Sensor Model Useful?

- Likelihood of a position (equivalent to conditional probability of obtaining measurement z given that the true value is x):

$$\mathscr{L}(x) = p(z \mid x)$$

probability of z (given x), or, likelihood of x (given that z was observed)



- Now, let us find the 'most likely' pose:

$$\hat{x} = \arg\max_{x} \mathscr{L}(x)$$

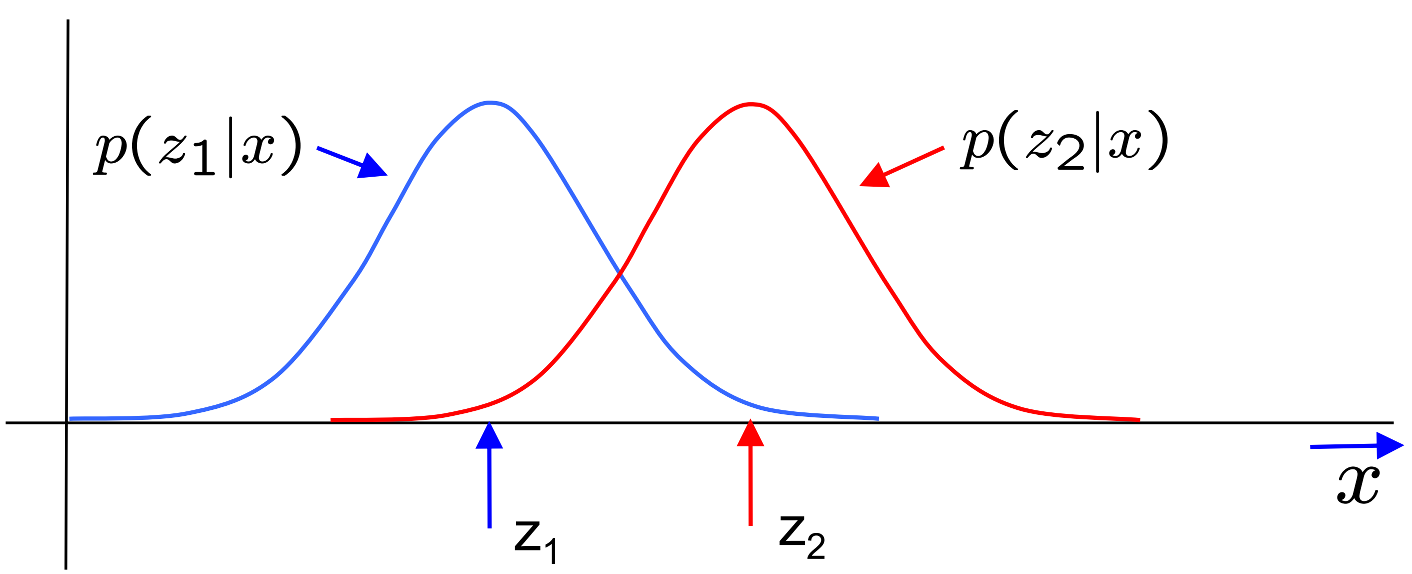
Maximum Likelihood Estimation

- Suppose we have two independent measurements z_1 and z_2 of position x, from two sensors with same variance. The sensor error is modeled as a Normal:

  $$p(z \mid x) = \mathcal{N}(x, \sigma^2)$$

- The likelihood is hence expressed as:

  $$\mathcal{L}(x) = p(z_1, z_2 \mid x) = p(z_1 \mid x)p(z_2 \mid x)$$

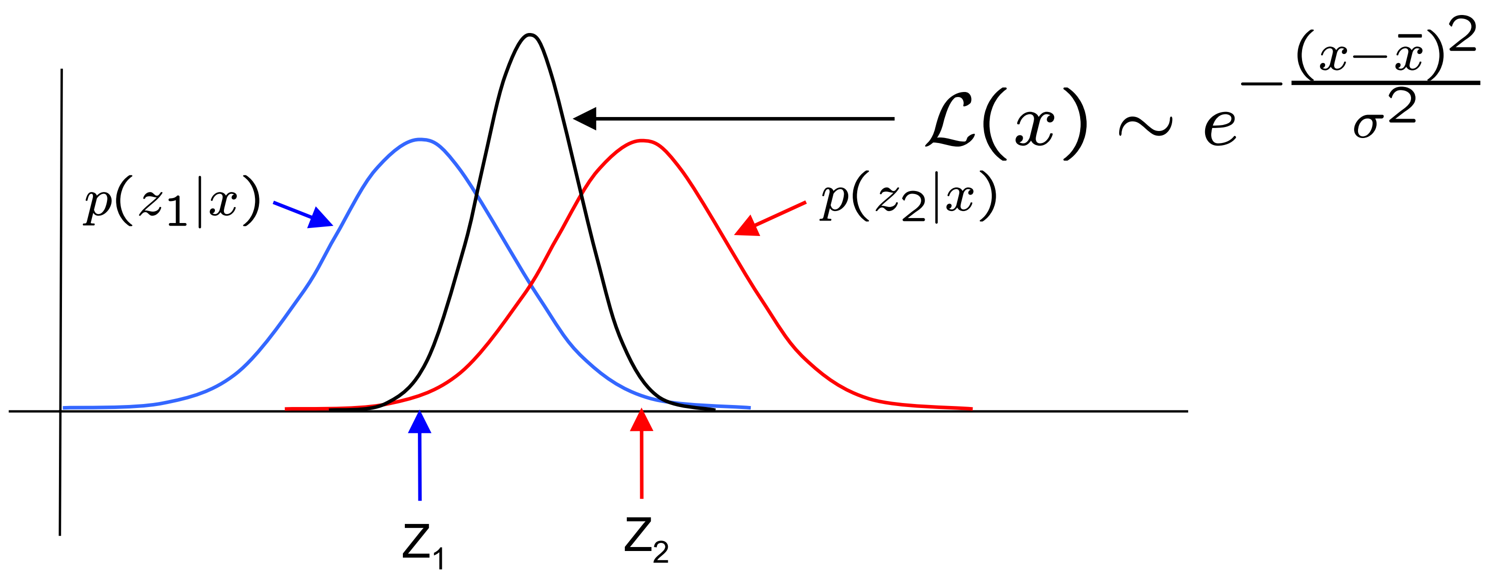


Maximum Likelihood Estimation

- We ignore normalization constants to compute:

$$\mathscr{L}(x) \sim e^{-\frac{(z_1 - x)^2}{2\sigma^2}} \times e^{-\frac{(z_2 - x)^2}{2\sigma^2}}$$

$$\mathscr{L}(x) \sim e^{-\frac{(x - \bar{x})^2}{\sigma^2}} \quad \text{with} \quad \bar{x} = \frac{z_1 + z_2}{2}$$



The likelihood is a Gaussian and the variance is reduced.

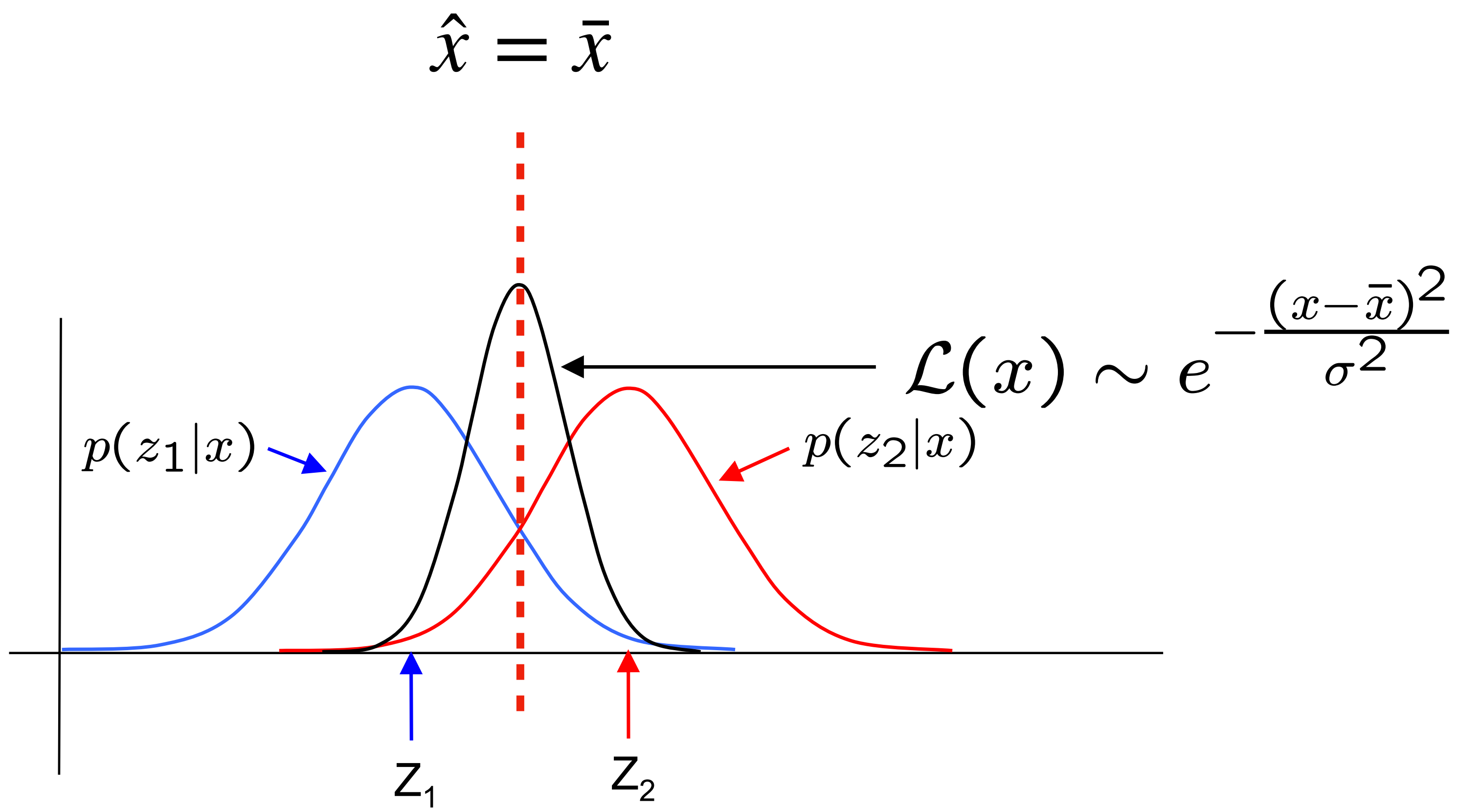
This is the foundation of **sensor fusion.**

Maximum Likelihood Estimation

- MLE trick: we minimize the negative log-likelihood, and then differentiate w.r.t. x:

$$\hat{x} = \arg\max_x \mathscr{L}(x) \longrightarrow \hat{x} = \arg\min_x \{-\ln \mathscr{L}(x)\}$$

$$\hat{x} = \bar{x} = \frac{z_1 + z_2}{2}$$



Further Reading

Books that cover fundamental concepts:

- Sensors for Mobile Robots, H. R. Everett, 1995
- Probabilistic Robotics, S. Thrun et al., 2006
- Elements of Robotics, F Mondada et al., 2018
- Autonomous Mobile Robots, R Siegwart et al., 2004