Lecture 21
Mobile Robotics - VI - Mapping

from Cyrill Stachniss

Course logistics

- Quiz 10 was posted yesterday noon and was due today noon.
- Project 7:
  - Groups are formed.
  - Two parts (~1 hr each) - Instructions will be prov
    1. Beginner's guide.
    2. Real Robot Challenge.
  - Scheduler is shared with the class.
    - Please book your 2 1-hour sessions
    - Both the parts needs to be completed by 04/

Location: Shepherd Labs 164 (Drone Lab) - this place will not be available for experiments after the dedicated times shown below.
Note: Talk to your team members and find a slot that works best to do P7 experiments. You will need two sessions as a team to perform the tasks we created for you. Please do not overbook. Start with 2 1-hour sessions.
You will need to come in as a team to finish these tasks.
Course staff will be present to guide you through the process.
Your Group Numbers are available in the next Sheet.

03/28/2024	04/01/2024	04/03/2024			04/04/2024		
		Robot-0	2:30-3:30 pm	Group-4	Robot-0	2:00-3:00 pm	Group-9
		Robot-1	2:30-3:30 pm	Group-2	Robot-1	2:00-3:00 pm	Available
		Robot 2	2:30 3:30 pm	Group 6	Robot 2	2:00 3:00 pm	Available
		Robot-3	2:30-3:30 pm	Group-11	Robot-3	2:00-3:00 pm	Available
		Robot-4	2:30-3:30 pm	Group-13	Robot-4	2:00-3:00 pm	Group-1
		Robot-0	3:30-4:30 pm	Group-10	Robot-0	3:00-4:00 pm	Group-3
		Robot-1	3:30-4:30 pm	Group-12	Robot-1	3:00-4:00 pm	Group-8
		Robot-2	3:30-4:30 pm	Group-6	Robot-2	3:00-4:00 pm	Available
		Robot-3	3:30-4:30 pm	Available	Robot-3	3:00-4:00 pm	Available
		Robot-4	3:30-4:30 pm	Group-7	Robot-4	3:00-4:00 pm	Available

04/08/2024			04/09/2024			04/11/2024			04/15/2024		
Robot-0	2:30-3:30 pm	Available	Robot-0	2:00-3:00 pm	Available	Robot-0	2:00-3:00 pm	Available	Robot-0	2:30-3:30 pm	Available
Robot-1	2:30-3:30 pm	Available	Robot-1	2:00-3:00 pm	Available	Robot-1	2:00-3:00 pm	Available	Robot-1	2:30-3:30 pm	Available
Robot-2	2:30-3:30 pm	Group-5	Robot-2	2:00-3:00 pm	Available	Robot-2	2:00-3:00 pm	Available	Robot-2	2:30-3:30 pm	Available
Robot 3	2:30-3:30 pm	Available	Robot 3	2:00-3:00 pm	Available	Robot 3	2:00-3:00 pm	Available	Robot 3	2:30-3:30 pm	Available
Robot-4	2:30-3:30 pm	Available	Robot-4	2:00-3:00 pm	Available	Robot-4	2:00-3:00 pm	Available	Robot-4	2:30-3:30 pm	Available
Robot-0	3:30-4:30 pm	Available	Robot-0	3:00-4:00 pm	Available	Robot-0	3:00-4:00 pm	Available	Robot-0	3:30-4:30 pm	Available
Robot-1	3:30-4:30 pm	Available	Robot-1	3:00-4:00 pm	Group-7	Robot-1	3:00-4:00 pm	Group-7	Robot-1	3:30-4:30 pm	Available
Robot-2	3:30-4:30 pm	Available	Robot-2	3:00-4:00 pm	Available	Robot-2	3:00-4:00 pm	Available	Robot-2	3:30-4:30 pm	Available
Robot-3	3:30-4:30 pm	Available	Robot-3	3:00-4:00 pm	Available	Robot-3	3:00-4:00 pm	Available	Robot-3	3:30-4:30 pm	Available
Robot-4	3:30-4:30 pm	Available	Robot-4	3:00-4:00 pm	Available	Robot-4	3:00-4:00 pm	Available	Robot-4	3:30-4:30 pm	Available

- No TA OHs between 03/28 and 04/12.
  - Karthik's OH will be available to discuss final projects.
  - Chahyon and Xun's OH are cancelled between 03/28 and 04/12. They maybe available upon request for the UNITE team.
- **Final Poster Session: 05/04/2024 - Saturday - 1pm - 4pm, Shepherd Labs 164 - mark your calendars**

Final (Open) Project timeline

Sun	Mon	Tue	Wed	Thu	Fri	Sat
31	1	2	Today 3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	1	2	3	4

In-class proposal presentation

Proposal Slides Due before the class

Poster for printing due

Poster Presentation

Final video due

Final (Open) Project timeline

- Proposal Slides: (template will be provided by 04/03)
  - 1-4 Slides
  - Title, Motivation, Input - Output, Evaluation, Deliverables, Timeline, Who is doing what?
  - Where does your project stand not the 3-axes (robots, objects, tasks)?
  - Backup plan
- In-class proposal presentation (<8mins) :
  - Teams will get feedback from the class
- Final video:
  - Describing the project idea and the outcome.
- Poster presentation: (template will be provided by 04/03)
  - Presenting the project idea and the outcome to audience.

- Final Project: 15%
  - Project proposal slides + presentation: 2%
  - Final project video: 5%
  - Poster presentation (evaluation by judges): 3%

Final Project (Open ended)

Think along these axes to decide your final project!

Evaluating your implementation/system with quantitative results are VERY important!

Objects

Rearrangment of a set of objects

You may use:

- Kineval codebase
- Other sim environments (pybullet, Gazebo, DRAKE, Isaac sim)
- Turtlebot3 (provided only upon compelling proposal, only 5 are available)
- Other robots you may have access to.

Multi-robot task execution

Robots

Long horizon tasks

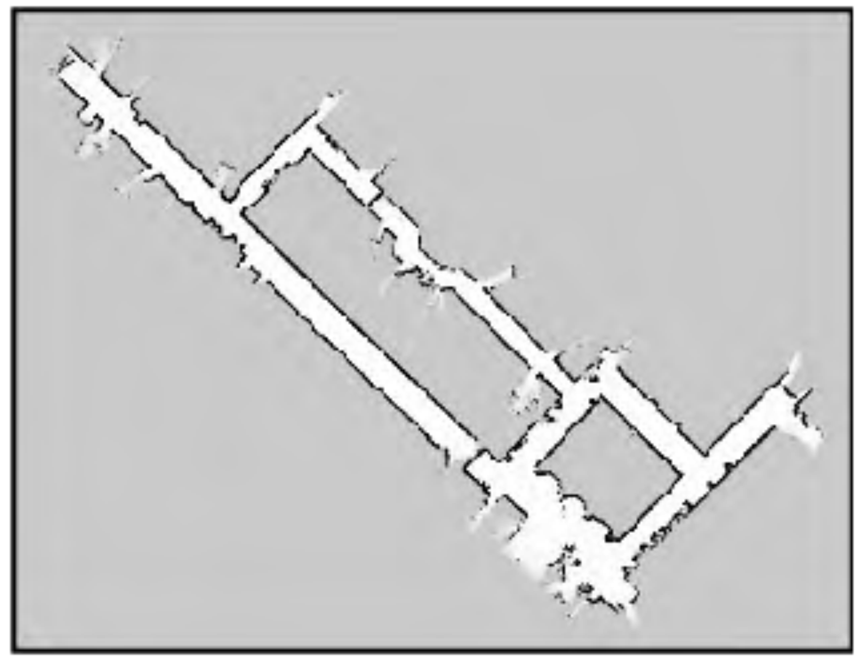
Tasks

Why Mapping?

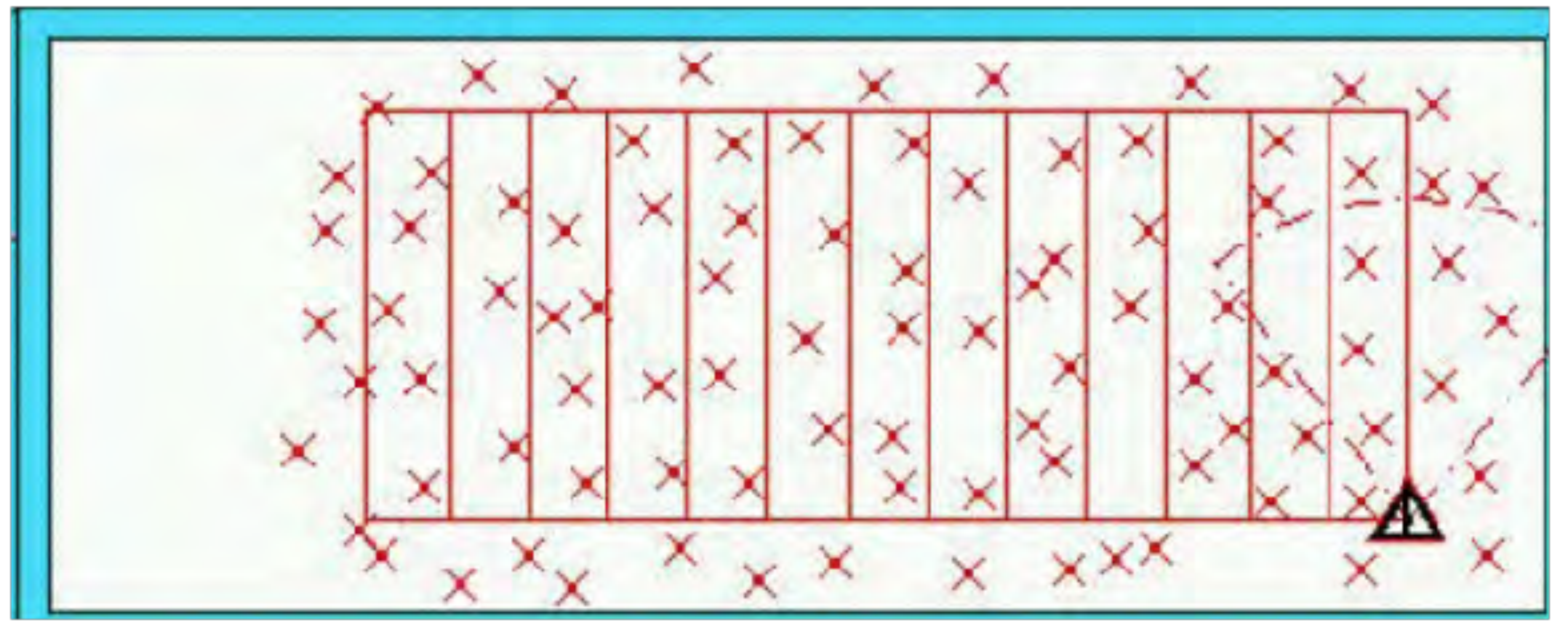
- Learning maps is one of the fundamental problems in mobile robotics
- Maps allow robots to efficiently carry out their tasks, allow localization …
- Successful robot systems rely on maps for localization, path planning, activity planning etc.

Types of Maps

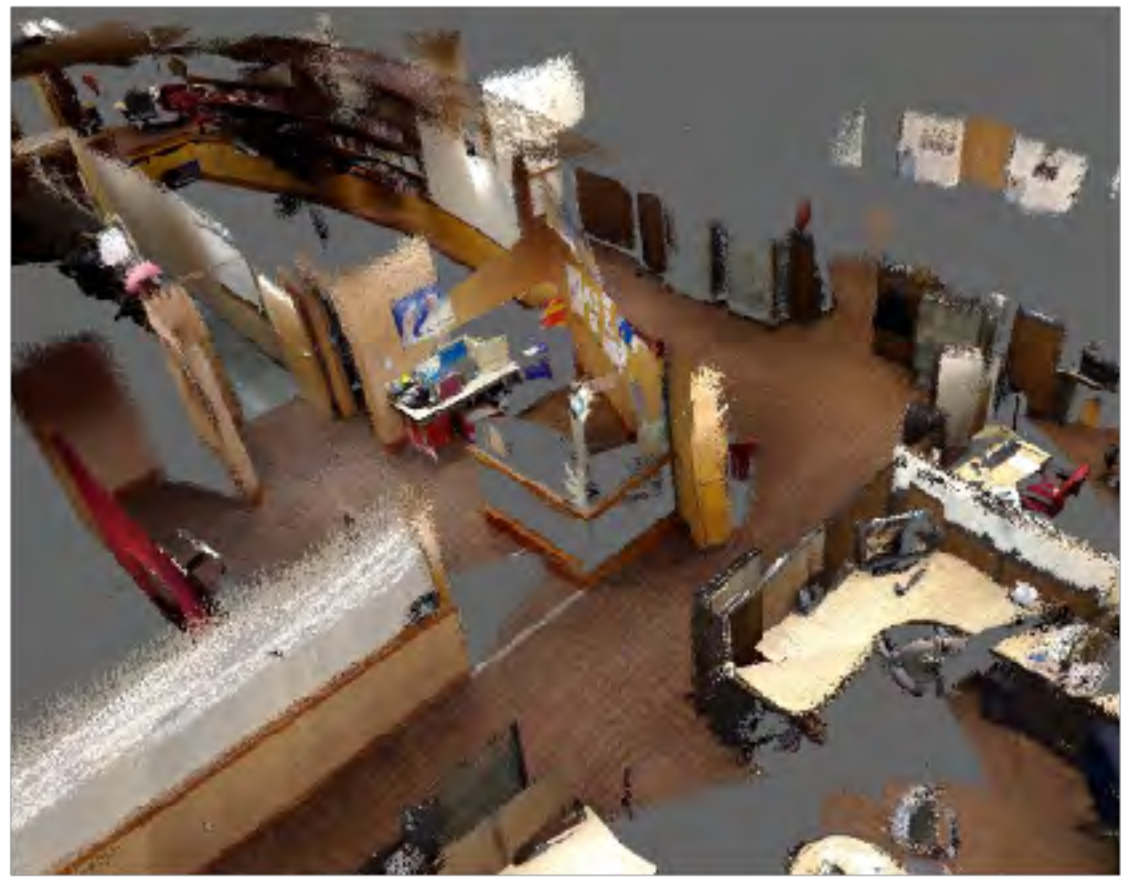
Grid maps or scans



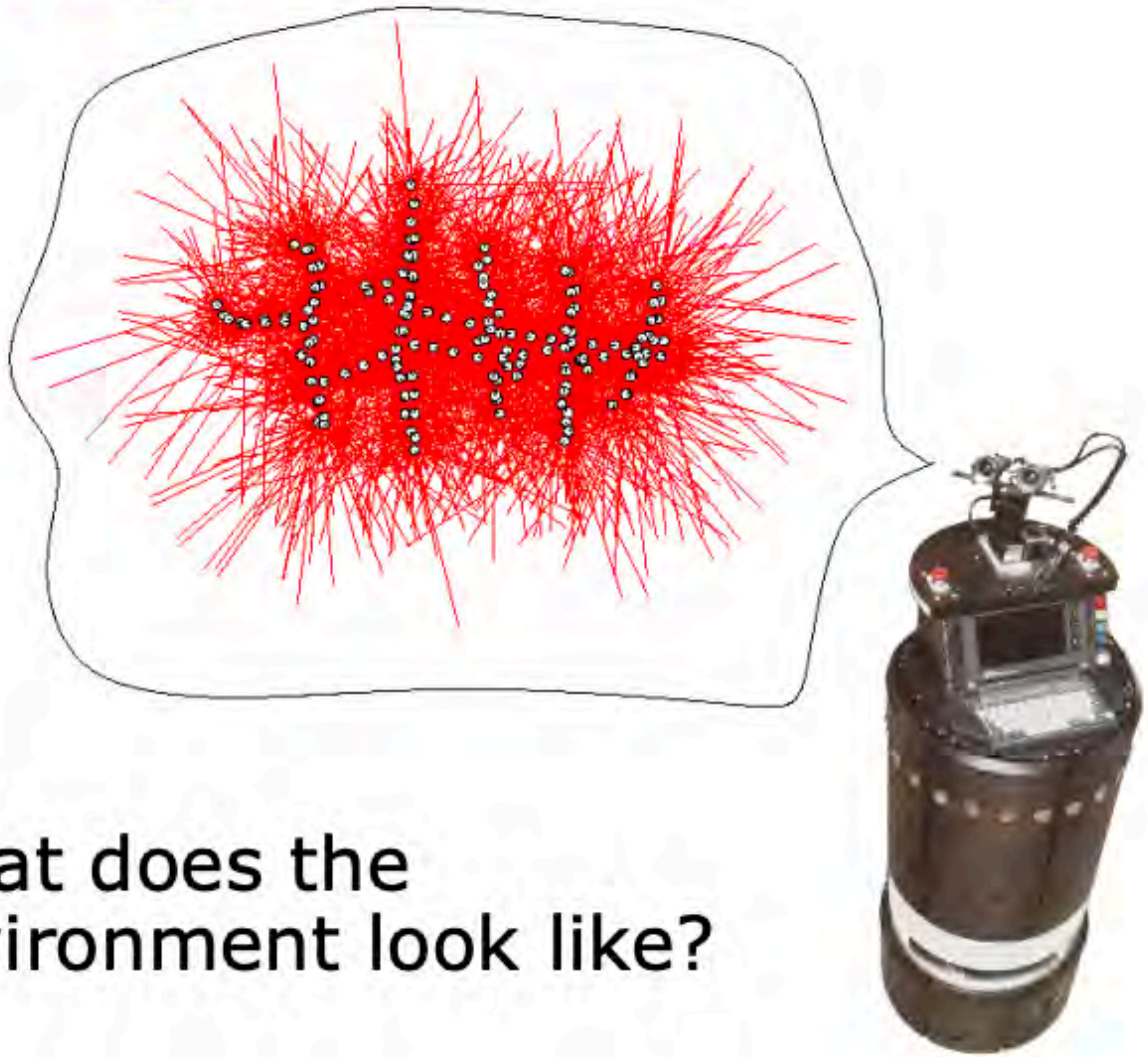
Sparse landmarks



RGB / Depth Maps



The General Problem of Mapping



What does the environment look like?

The General Problem of Mapping

- Formally, mapping involves, given the sensor data,

$$d = \{u_1, z_1, u_2, z_2, \ldots, u_n, z_n\}$$

to calculate the most likely map

$$m^* = \underset{m}{\arg\max}\, P(m \mid d)$$

Mapping as a Chicken and Egg Problem

- So far we learned how to estimate the pose of the vehicle given the data and the map.

Mapping as a Chicken and Egg Problem

- So far we learned how to estimate the pose of the vehicle given the data and the map.
- Mapping, however, involves to simultaneously estimate the pose of the vehicle and the map.
- The general problem is therefore denoted as the simultaneous localization and mapping problem (SLAM).
- Throughout this section we will describe how to calculate a map given we know the pose of the vehicle.

Problems in Mapping

- Sensor interpretation
  - How do we extract relevant information from raw sensor data?
  - How do we represent and integrate this information over time?

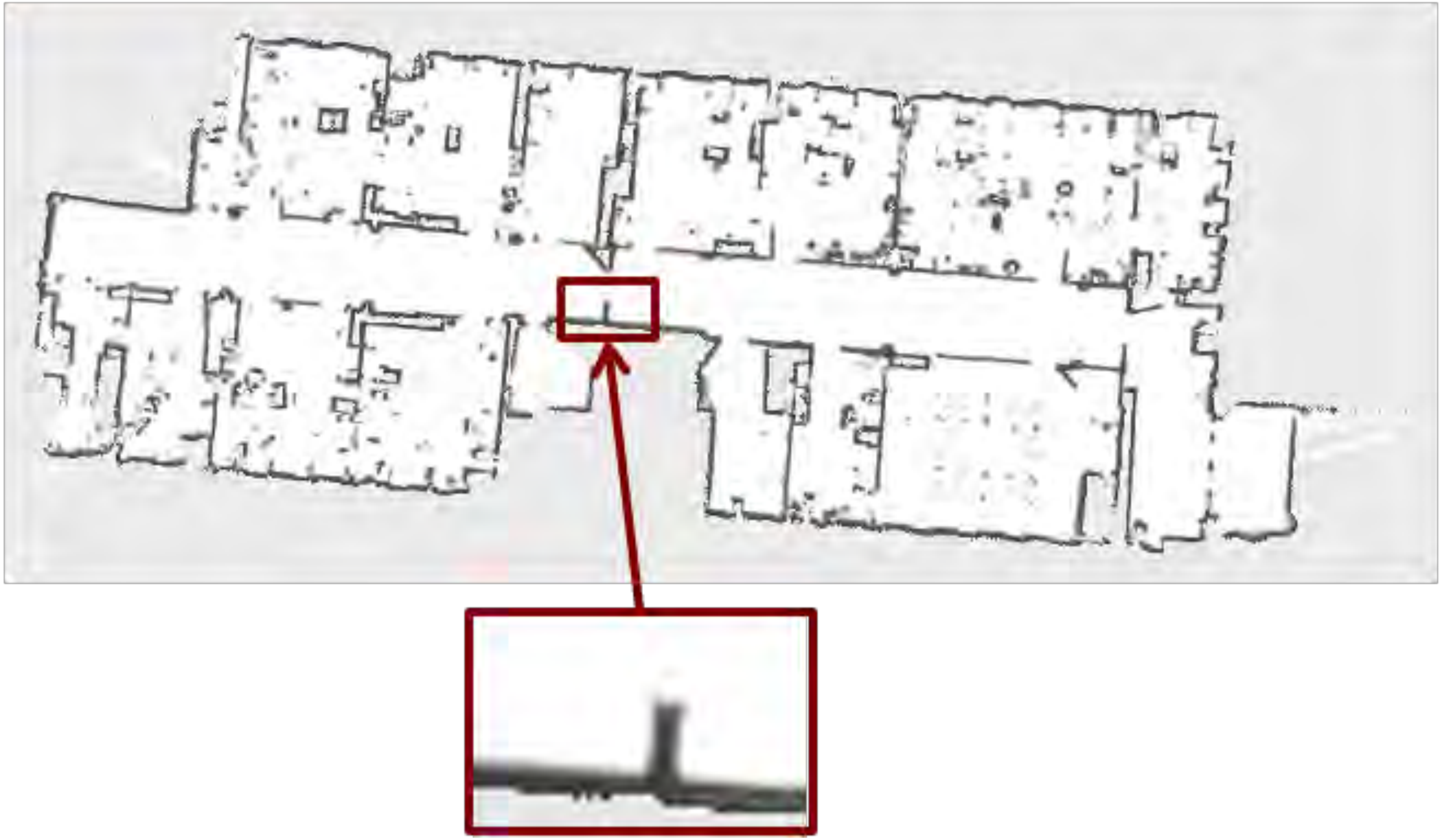
- Robot locations have to be known
  - How can we estimate them during mapping?

Occupancy Grid Mapping

Grid Maps

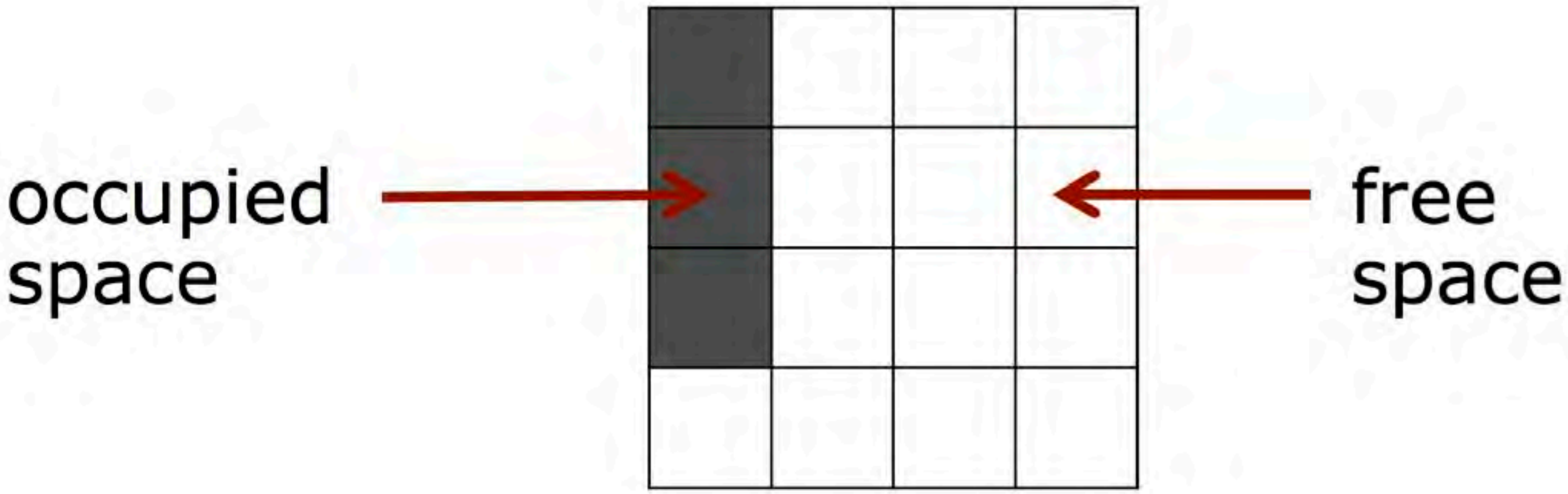
- Discretize the world into cells
- Grid structure is rigid
- Each cell is assumed to be occupied or free space
- Non-parametric model
- Large maps require substantial memory resources
- Do not rely on a feature detector

Example



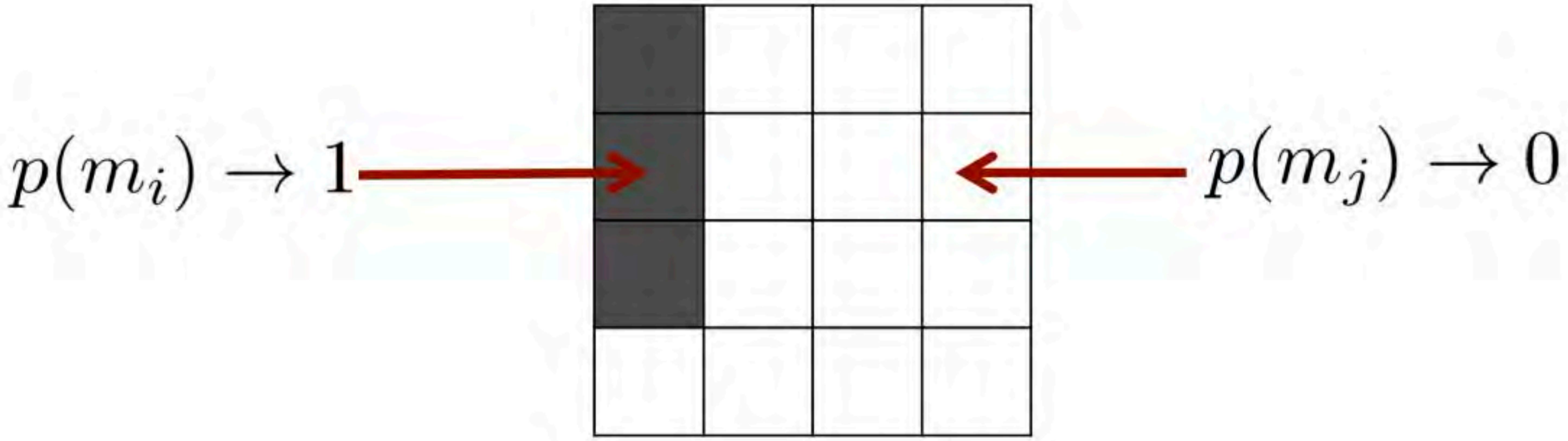
Assumption 1

- The area that corresponds to a cell is either completely free or occupied



Representation

- Each cell is a **binary random variable** that models the occupancy

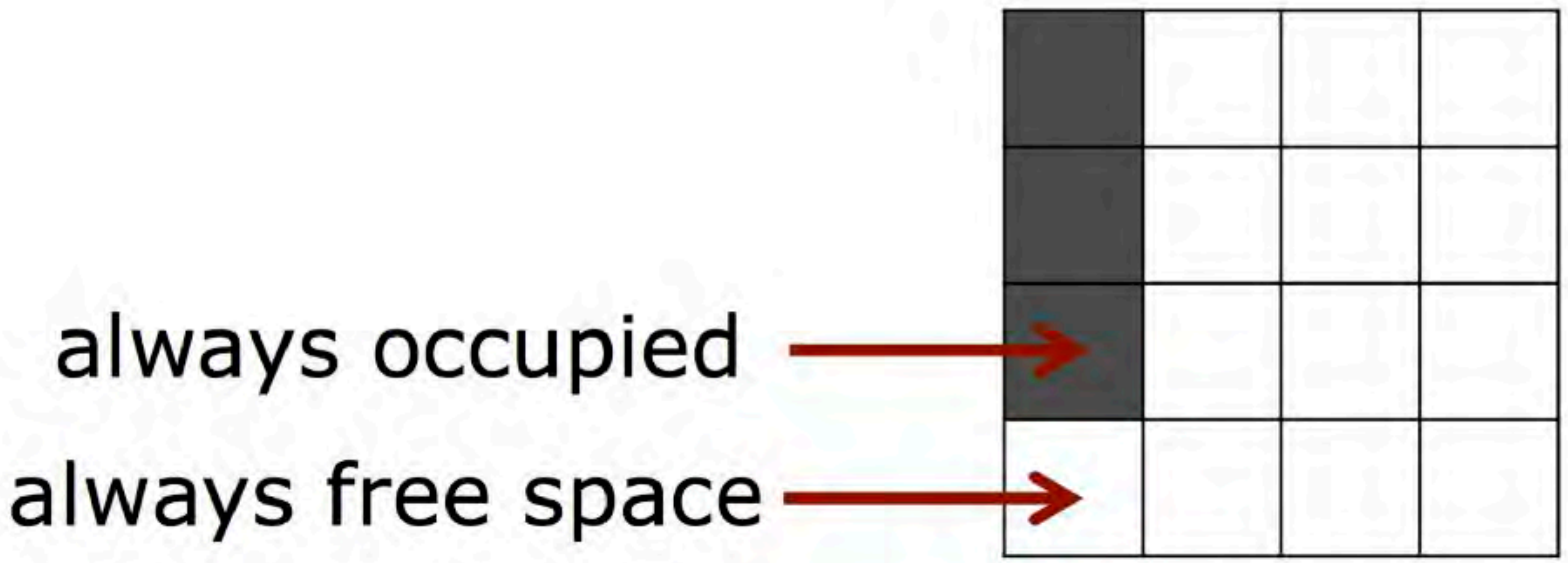


Occupancy Probability

- Each cell is a **binary random variable** that models the occupancy
- Cell is occupied: $p(m_i) = 1$
- Cell is not occupied: $p(m_i) = 0$
- No knowledge: $p(m_i) = 0.5$

Assumption 2

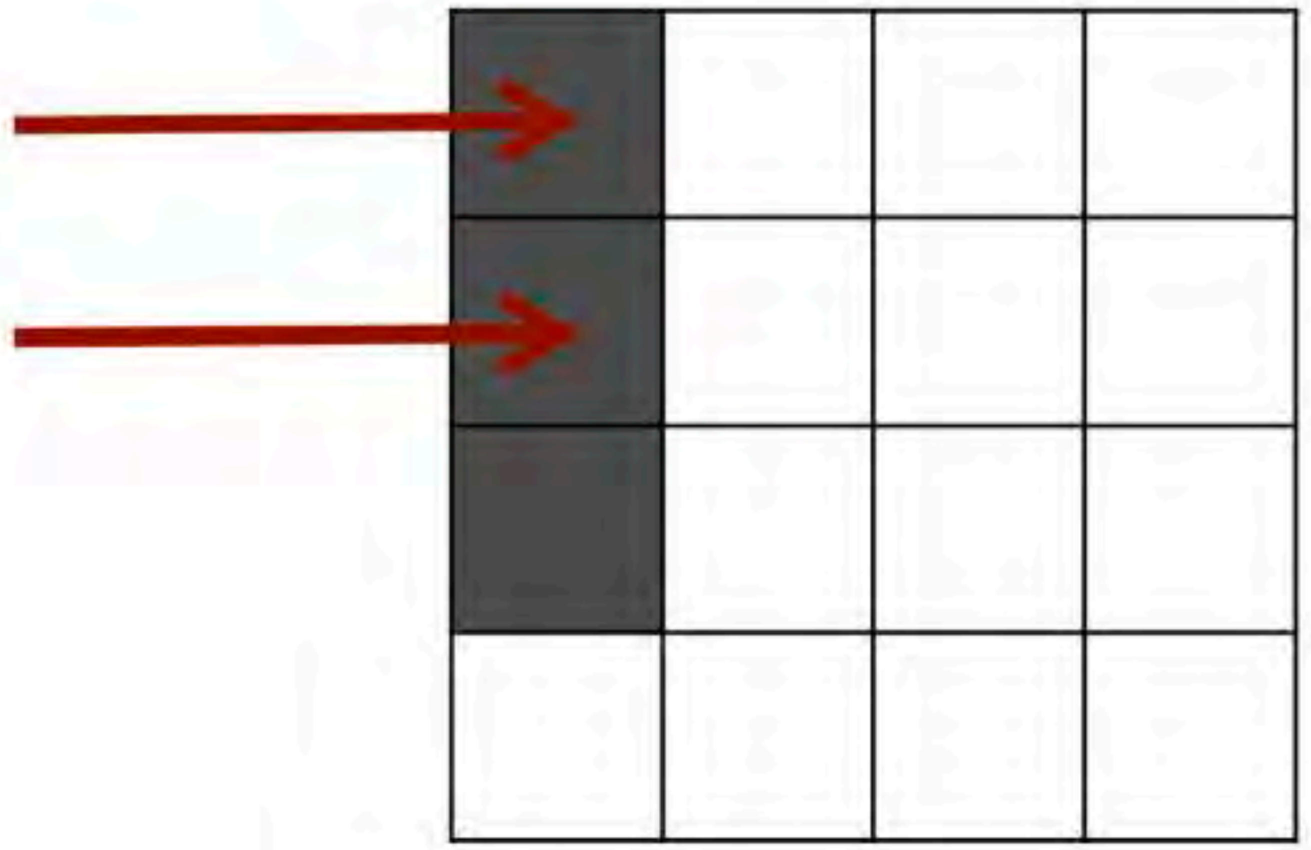
- The world is **static** (most mapping systems make this assumption)



Assumption 3

- The cells (the random variables) are **independent** of each other

no dependency
between the cells



Representation

- The probability distribution of the map is given by the product over the cells

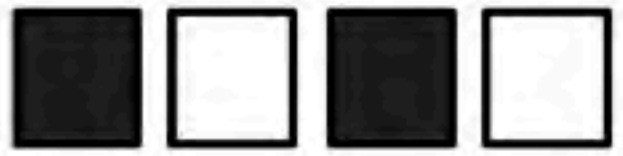
$$\underbrace{p(m)}_{\text{map}} = \prod_i \underbrace{p(m_i)}_{\text{cell}}$$

Representation

- The probability distribution of the map is given by the product over the cells

$$p(m) = \prod_i p(m_i)$$

example map
(4-dim state)



4 individual cells

Estimating a Map From Data

- Given sensor data $z_{1:t}$ and the poses $x_{1:t}$ of the sensor, estimate the map

$$p(m \mid z_{1:t}, x_{1:t}) \quad = \quad \prod_i p(m_i \mid z_{1:t}, x_{1:t})$$

binary random variable

➡ Binary Bayes filter (for a static state)

Static State Binary Bayes Filter

$$p(m_i \mid z_{1:t}, x_{1:t}) \overset{\text{Bayes rule}}{=} \frac{p(z_t \mid m_i, z_{1:t-1}, x_{1:t})\, p(m_i \mid z_{1:t-1}, x_{1:t})}{p(z_t \mid z_{1:t-1}, x_{1:t})}$$

Static State Binary Bayes Filter

$$p(m_i \mid z_{1:t}, x_{1:t}) \overset{\text{Bayes rule}}{=} \frac{p(z_t \mid m_i, z_{1:t-1}, x_{1:t})\, p(m_i \mid z_{1:t-1}, x_{1:t})}{p(z_t \mid z_{1:t-1}, x_{1:t})}$$

$$\overset{\text{Markov}}{=} \frac{p(z_t \mid m_i, x_t)\, p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(z_t \mid z_{1:t-1}, x_{1:t})}$$

Static State Binary Bayes Filter

$$p(m_i \mid z_{1:t}, x_{1:t}) \overset{\text{Bayes rule}}{=} \frac{p(z_t \mid m_i, z_{1:t-1}, x_{1:t})\, p(m_i \mid z_{1:t-1}, x_{1:t})}{p(z_t \mid z_{1:t-1}, x_{1:t})}$$

$$\overset{\text{Markov}}{=} \frac{p(z_t \mid m_i, x_t)\, p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(z_t \mid z_{1:t-1}, x_{1:t})}$$

$$p(z_t \mid m_i, x_t) \overset{\text{Bayes rule}}{=} \frac{p(m_i \mid z_t, x_t)\, p(z_t \mid x_t)}{p(m_i \mid x_t)}$$

Static State Binary Bayes Filter

$$
\begin{aligned}
p(m_i \mid z_{1:t}, x_{1:t}) &\overset{\text{Bayes rule}}{=} \frac{p(z_t \mid m_i, z_{1:t-1}, x_{1:t})\, p(m_i \mid z_{1:t-1}, x_{1:t})}{p(z_t \mid z_{1:t-1}, x_{1:t})} \\
&\overset{\text{Markov}}{=} \frac{p(z_t \mid m_i, x_t)\, p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(z_t \mid z_{1:t-1}, x_{1:t})} \\
&\overset{\text{Bayes rule}}{=} \frac{p(m_i \mid z_t, x_t)\, p(z_t \mid x_t)\, p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(m_i \mid x_t)\, p(z_t \mid z_{1:t-1}, x_{1:t})}
\end{aligned}
$$

Static State Binary Bayes Filter

$$
\begin{aligned}
p(m_i \mid z_{1:t}, x_{1:t}) &\overset{\text{Bayes rule}}{=} \frac{p(z_t \mid m_i, z_{1:t-1}, x_{1:t})\, p(m_i \mid z_{1:t-1}, x_{1:t})}{p(z_t \mid z_{1:t-1}, x_{1:t})} \\
&\overset{\text{Markov}}{=} \frac{p(z_t \mid m_i, x_t)\, p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(z_t \mid z_{1:t-1}, x_{1:t})} \\
&\overset{\text{Bayes rule}}{=} \frac{p(m_i \mid z_t, x_t)\, p(z_t \mid x_t)\, p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(m_i \mid x_t)\, p(z_t \mid z_{1:t-1}, x_{1:t})} \\
&\overset{\text{Markov}}{=} \frac{p(m_i \mid z_t, x_t)\, p(z_t \mid x_t)\, p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(m_i)\, p(z_t \mid z_{1:t-1}, x_{1:t})}
\end{aligned}
$$

Static State Binary Bayes Filter

$$p(m_i \mid z_{1:t}, x_{1:t}) \stackrel{\text{Bayes rule}}{=} \frac{p(z_t \mid m_i, z_{1:t-1}, x_{1:t})\, p(m_i \mid z_{1:t-1}, x_{1:t})}{p(z_t \mid z_{1:t-1}, x_{1:t})}$$

$$\stackrel{\text{Markov}}{=} \frac{p(z_t \mid m_i, x_t)\, p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(z_t \mid z_{1:t-1}, x_{1:t})}$$

$$\stackrel{\text{Bayes rule}}{=} \frac{p(m_i \mid z_t, x_t)\, p(z_t \mid x_t)\, p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(m_i \mid x_t)\, p(z_t \mid z_{1:t-1}, x_{1:t})}$$

$$\stackrel{\text{Markov}}{=} \frac{p(m_i \mid z_t, x_t)\, p(z_t \mid x_t)\, p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(m_i)\, p(z_t \mid z_{1:t-1}, x_{1:t})}$$

Do exactly the same for the opposite event:

$$p(\neg m_i \mid z_{1:t}, x_{1:t}) \stackrel{\text{the same}}{=} \frac{p(\neg m_i \mid z_t, x_t)\, p(z_t \mid x_t)\, p(\neg m_i \mid z_{1:t-1}, x_{1:t-1})}{p(\neg m_i)\, p(z_t \mid z_{1:t-1}, x_{1:t})}$$

Static State Binary Bayes Filter

- By computing the ratio of both probabilities, we obtain:

$$\frac{p(m_i \mid z_{1:t}, x_{1:t})}{p(\neg m_i \mid z_{1:t}, x_{1:t})} = \frac{\frac{p(m_i|z_t,x_t)\,\cancel{p(z_t|x_t)}\,p(m_i|z_{1:t-1},x_{1:t-1})}{p(m_i)\,\cancel{p(z_t|z_{1:t-1},x_{1:t})}}}{\frac{p(\neg m_i|z_t,x_t)\,\cancel{p(z_t|x_t)}\,p(\neg m_i|z_{1:t-1},x_{1:t-1})}{p(\neg m_i)\,\cancel{p(z_t|z_{1:t-1},x_{1:t})}}}$$

Static State Binary Bayes Filter

- By computing the ratio of both probabilities, we obtain:

$$
\begin{aligned}
&\frac{p(m_i \mid z_{1:t}, x_{1:t})}{p(\neg m_i \mid z_{1:t}, x_{1:t})} \\
&\quad = \frac{p(m_i \mid z_t, x_t)\, p(m_i \mid z_{1:t-1}, x_{1:t-1})\, p(\neg m_i)}{p(\neg m_i \mid z_t, x_t)\, p(\neg m_i \mid z_{1:t-1}, x_{1:t-1})\, p(m_i)} \\
&\quad = \frac{p(m_i \mid z_t, x_t)}{1 - p(m_i \mid z_t, x_t)} \frac{p(m_i \mid z_{1:t-1}, x_{1:t-1})}{1 - p(m_i \mid z_{1:t-1}, x_{1:t-1})} \frac{1 - p(m_i)}{p(m_i)}
\end{aligned}
$$

Static State Binary Bayes Filter

- By computing the ratio of both probabilities, we obtain:

$$\frac{p(m_i \mid z_{1:t}, x_{1:t})}{1 - p(m_i \mid z_{1:t}, x_{1:t})}$$

$$= \frac{p(m_i \mid z_t, x_t)\, p(m_i \mid z_{1:t-1}, x_{1:t-1})\, p(\neg m_i)}{p(\neg m_i \mid z_t, x_t)\, p(\neg m_i \mid z_{1:t-1}, x_{1:t-1})\, p(m_i)}$$

$$= \underbrace{\frac{p(m_i \mid z_t, x_t)}{1 - p(m_i \mid z_t, x_t)}}_{\text{uses } z_t} \underbrace{\frac{p(m_i \mid z_{1:t-1}, x_{1:t-1})}{1 - p(m_i \mid z_{1:t-1}, x_{1:t-1})}}_{\text{recursive term}} \underbrace{\frac{1 - p(m_i)}{p(m_i)}}_{\text{prior}}$$

From Ratio to Probability

We can turn the ratio into a probability:

$$
\begin{aligned}
\frac{p(x)}{1-p(x)} &= Y \\
p(x) &= Y - Y\, p(x) \\
p(x)\,(1+Y) &= Y \\
p(x) &= \frac{Y}{1+Y} \\
p(x) &= \frac{1}{1+\frac{1}{Y}}
\end{aligned}
$$

From Ratio to Probability

- Using $p(x) = [1 + Y^{-1}]^{-1}$ directly leads to

$$p(m_i \mid z_{1:t}, x_{1:t})$$
$$= \left[1 + \frac{1 - p(m_i \mid z_t, x_t)}{p(m_i \mid z_t, x_t)} \frac{1 - p(m_i \mid z_{1:t-1}, x_{1:t-1})}{p(m_i \mid z_{1:t-1}, x_{1:t-1})} \frac{p(m_i)}{1 - p(m_i)}\right]^{-1}$$

For reasons of efficiency, one performs the calculations in the log odds notation

Log Odds Notation

- The log odds notation computes the logarithm of the ratio of probabilities

$$\frac{p(m_i \mid z_{1:t}, x_{1:t})}{1 - p(m_i \mid z_{1:t}, x_{1:t})}$$

$$= \underbrace{\frac{p(m_i \mid z_t, x_t)}{1 - p(m_i \mid z_t, x_t)}}_{\text{uses } z_t} \underbrace{\frac{p(m_i \mid z_{1:t-1}, x_{1:t-1})}{1 - p(m_i \mid z_{1:t-1}, x_{1:t-1})}}_{\text{recursive term}} \underbrace{\frac{1 - p(m_i)}{p(m_i)}}_{\text{prior}}$$

$$\Rightarrow \quad l(m_i \mid z_{1:t}, x_{1:t}) = \log \left(\frac{p(m_i \mid z_{1:t}, x_{1:t})}{1 - p(m_i \mid z_{1:t}, x_{1:t})} \right)$$

Log Odds Notation

- Log odds ratio is defined as

$$l(x) \quad = \quad \log \frac{p(x)}{1 - p(x)}$$

- and with the ability to retrieve $p(x)$

$$p(x) \quad = \quad 1 - \frac{1}{1 + \exp l(x)}$$

Occupancy Mapping in Log Odds Form

- The product turns into a sum

$$l(m_i \mid z_{1:t}, x_{1:t}) = \underbrace{l(m_i \mid z_t, x_t)}_{\text{inverse sensor model}} + \underbrace{l(m_i \mid z_{1:t-1}, x_{1:t-1})}_{\text{recursive term}} - \underbrace{l(m_i)}_{\text{prior}}$$

- or in short

$$l_{t,i} = \text{inv\_sensor\_model}(m_i, x_t, z_t) + l_{t-1,i} - l_0$$

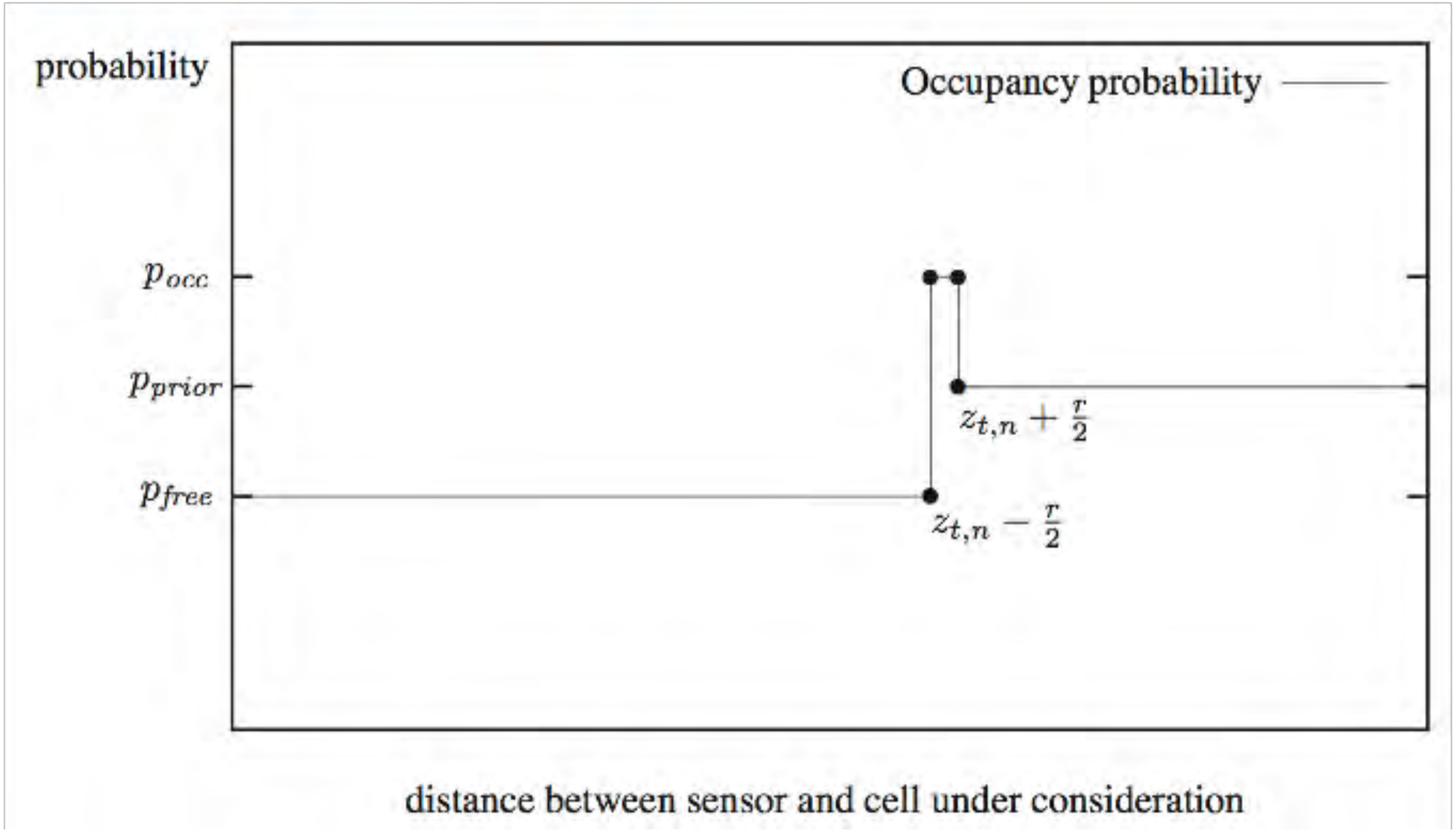
Occupancy Mapping Algorithm

occupancy_grid_mapping($\{l_{t-1,i}\}, x_t, z_t$)**:**

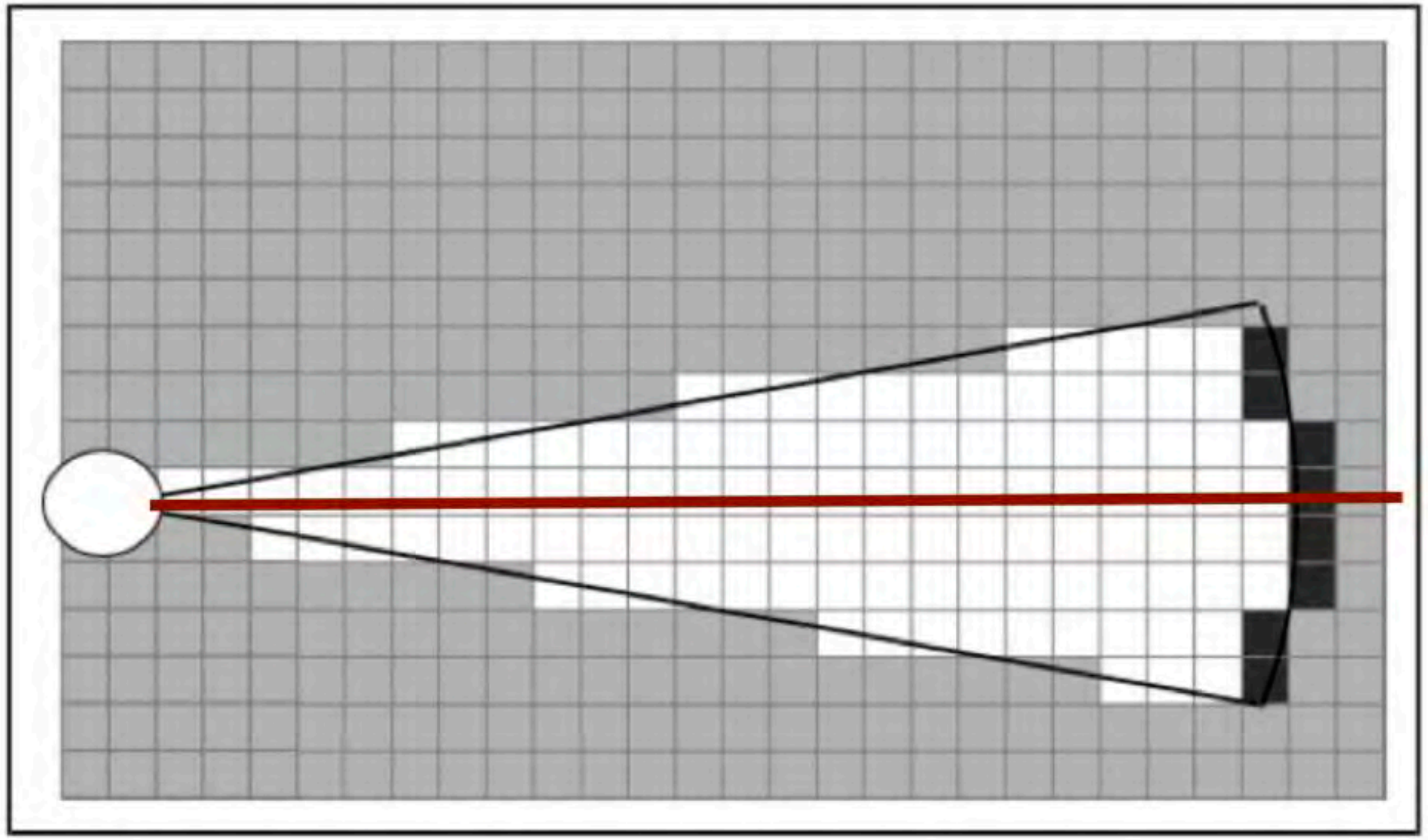
for all cells m_i *do*
  if m_i *in perceptual field of* z_t *then*
    $l_{t,i} = l_{t-1,i} + \text{inv\_sensor\_model}(m_i, x_t, z_t) - l_0$
  else
    $l_{t,i} = l_{t-1,i}$
  endif
endfor
return $\{l_{t,i}\}$

highly efficient, we only have to compute sums

Inverse Sensor Model for Laser Range Finders

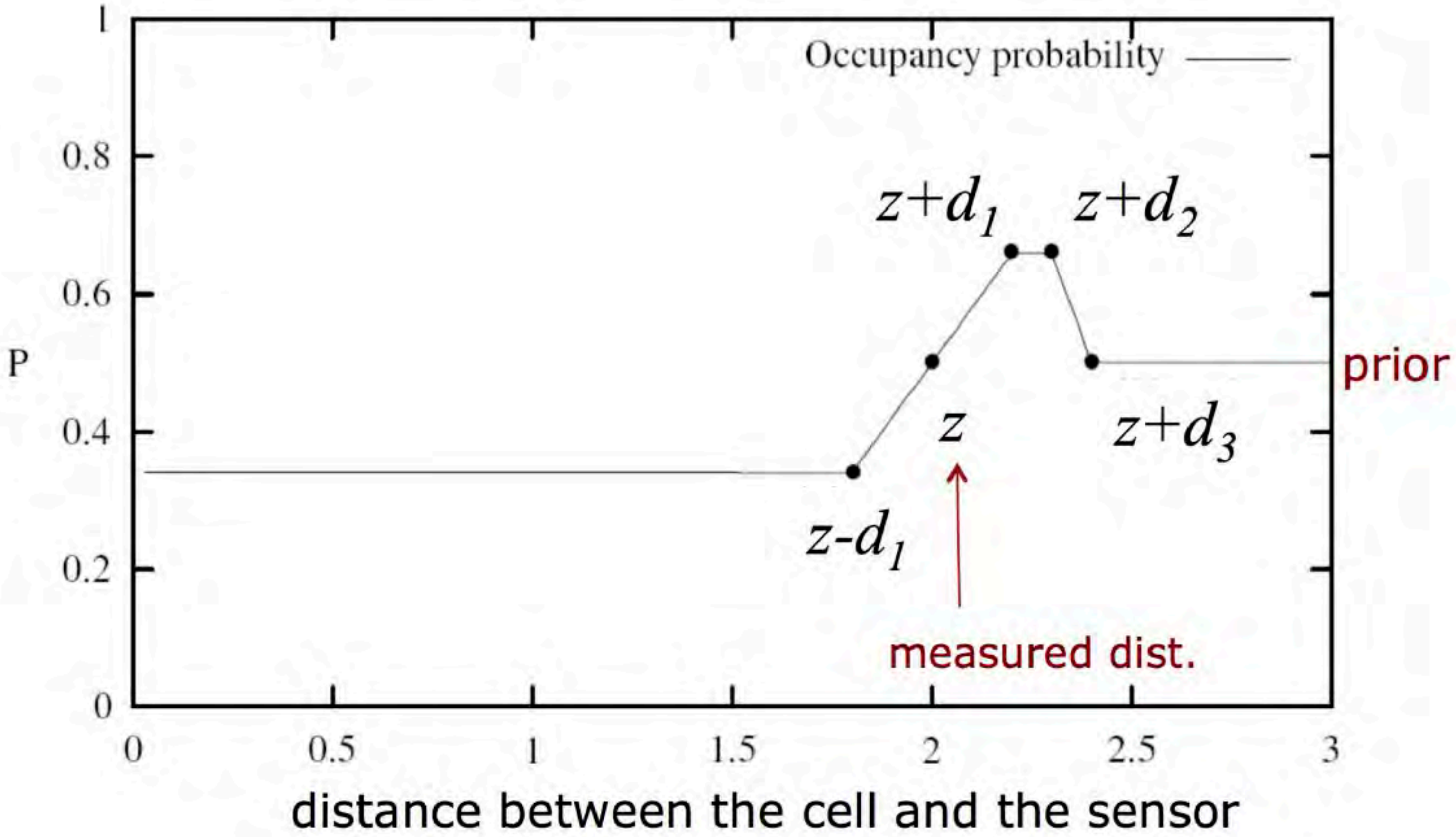


Inverse Sensor Model for Sonar Range Sensors

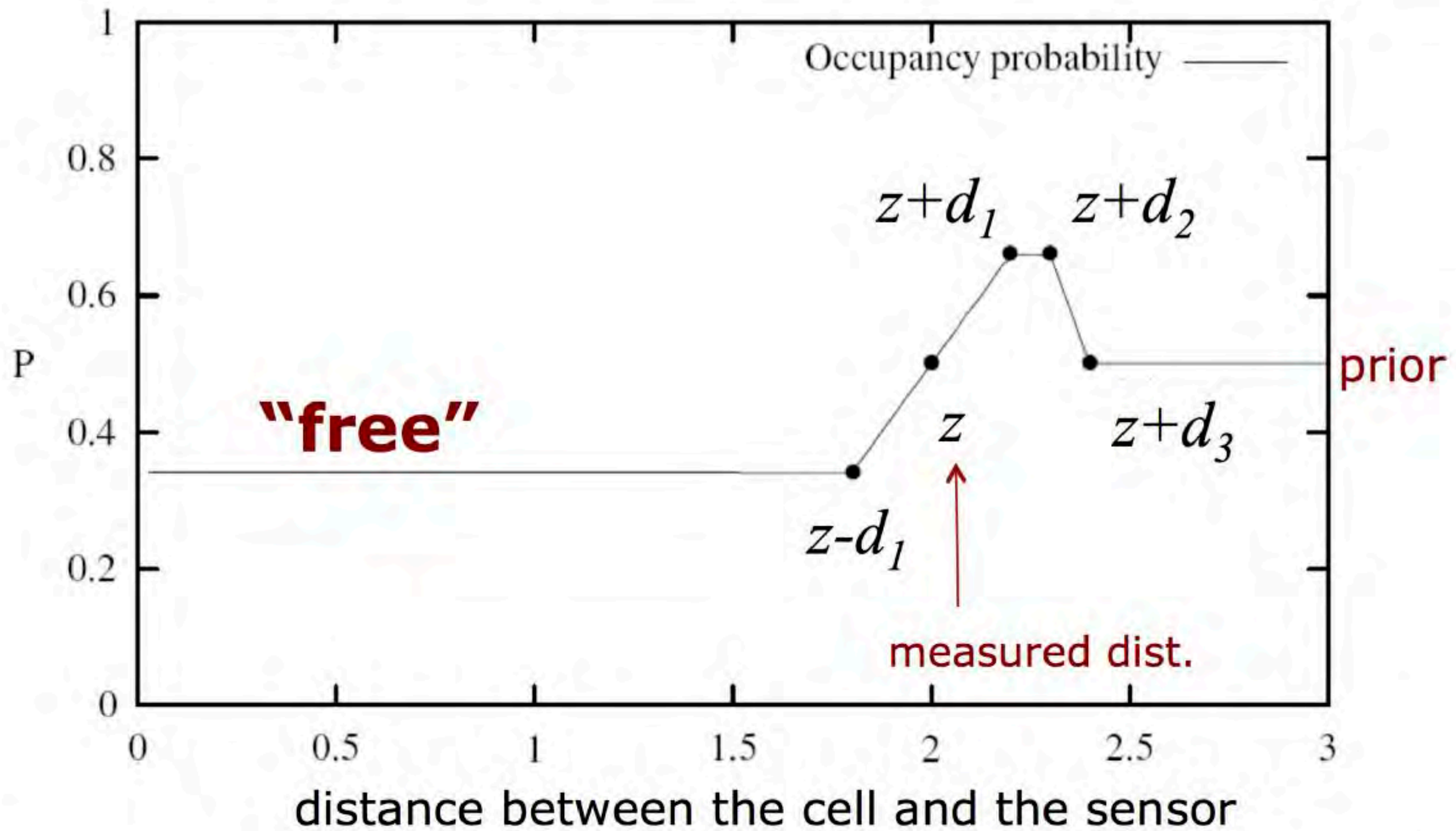


In the following, consider the cells along the optical axis (red line)

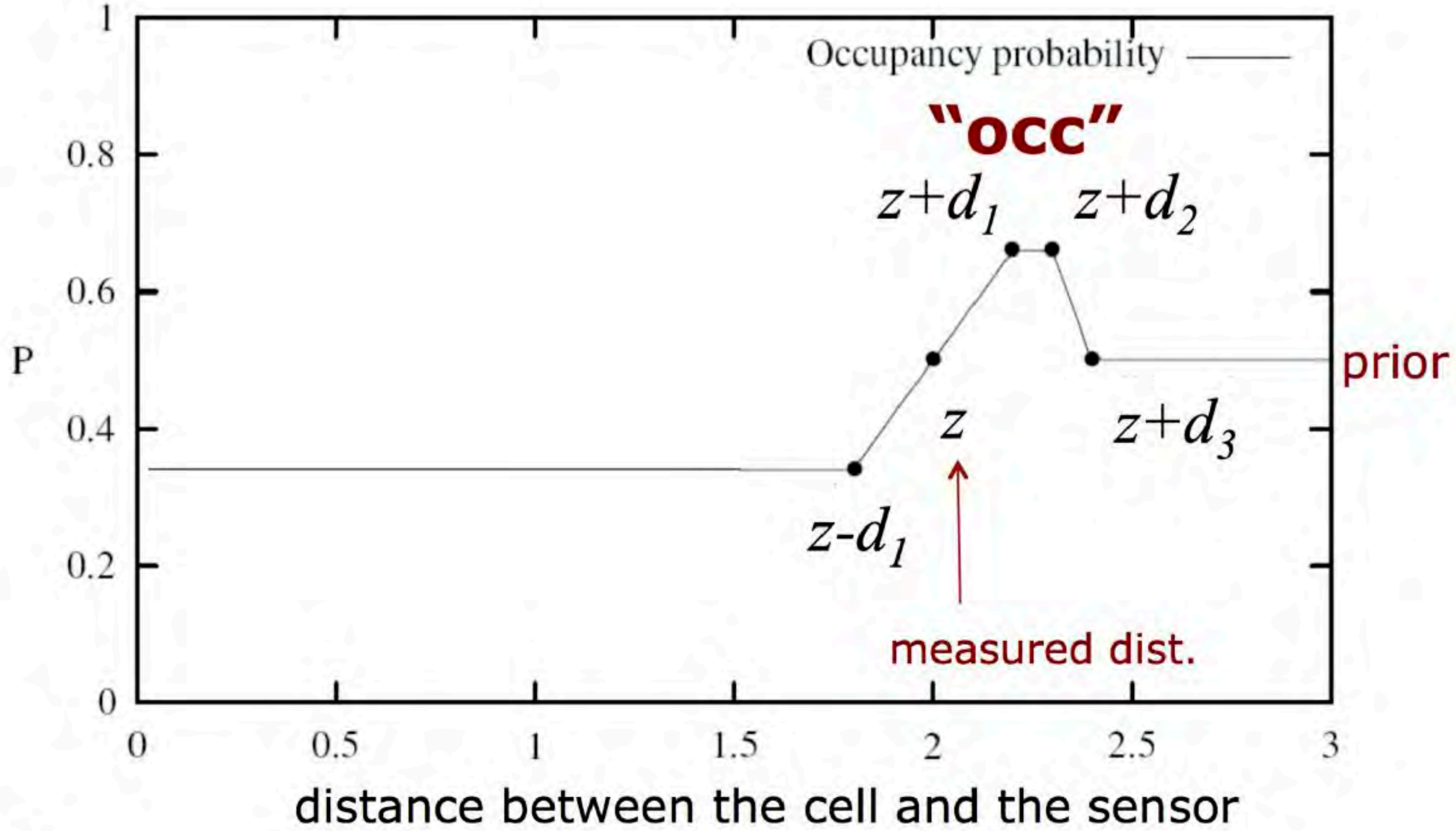
Occupancy Value Depending on the Measured Distance



Occupancy Value Depending on the Measured Distance

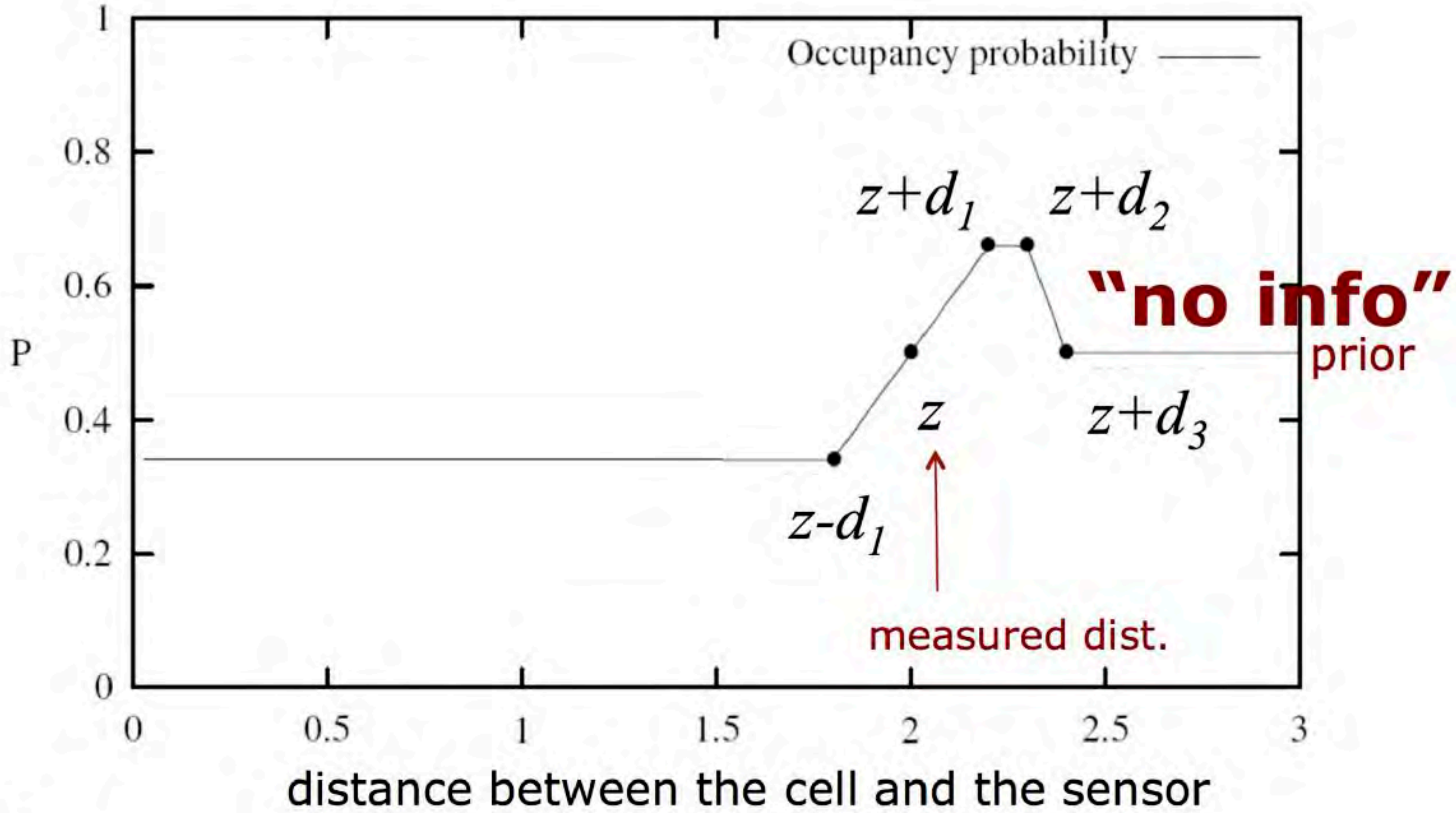


Occupancy Value Depending on the Measured Distance

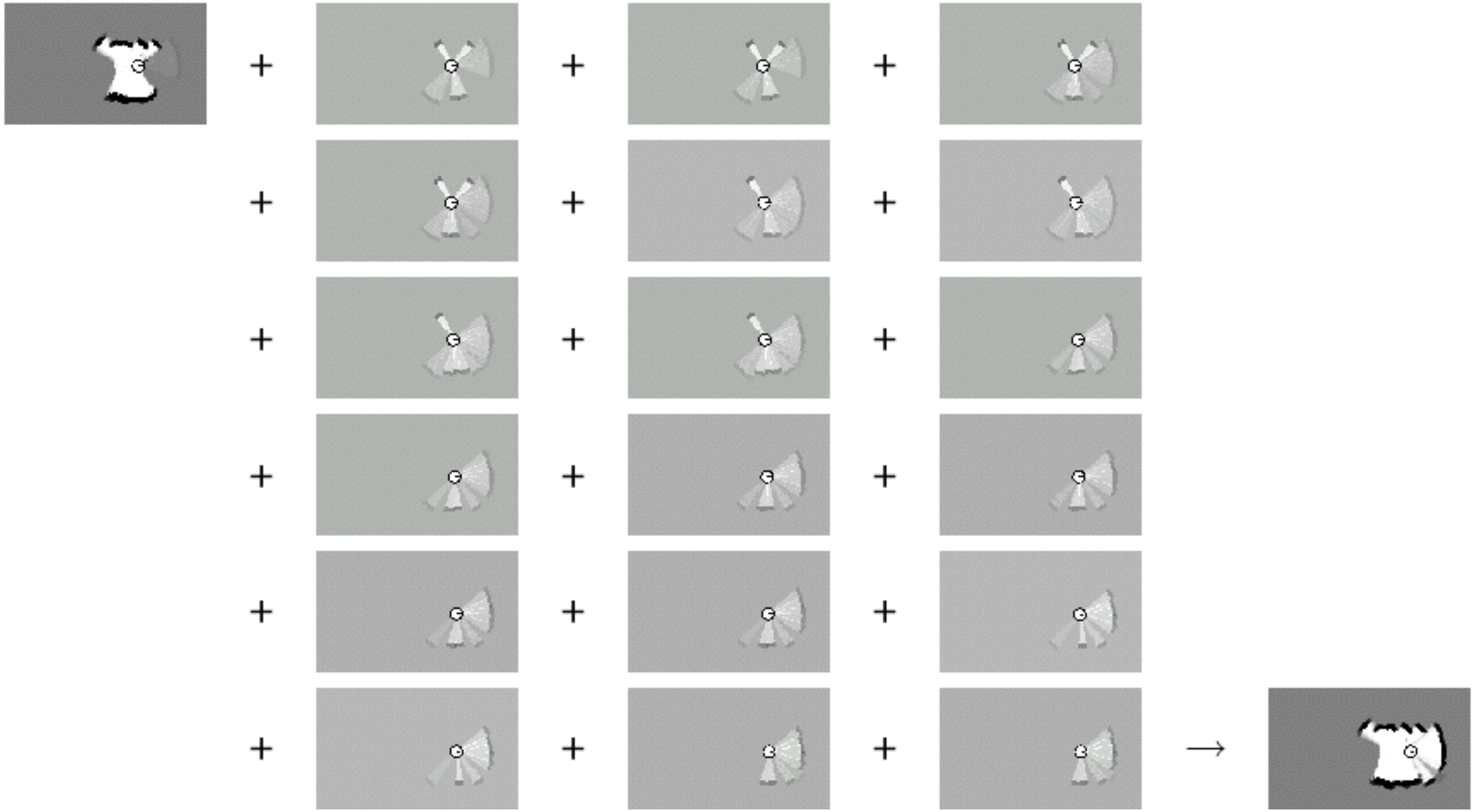


Slide borrowed from Cyrill Stachniss

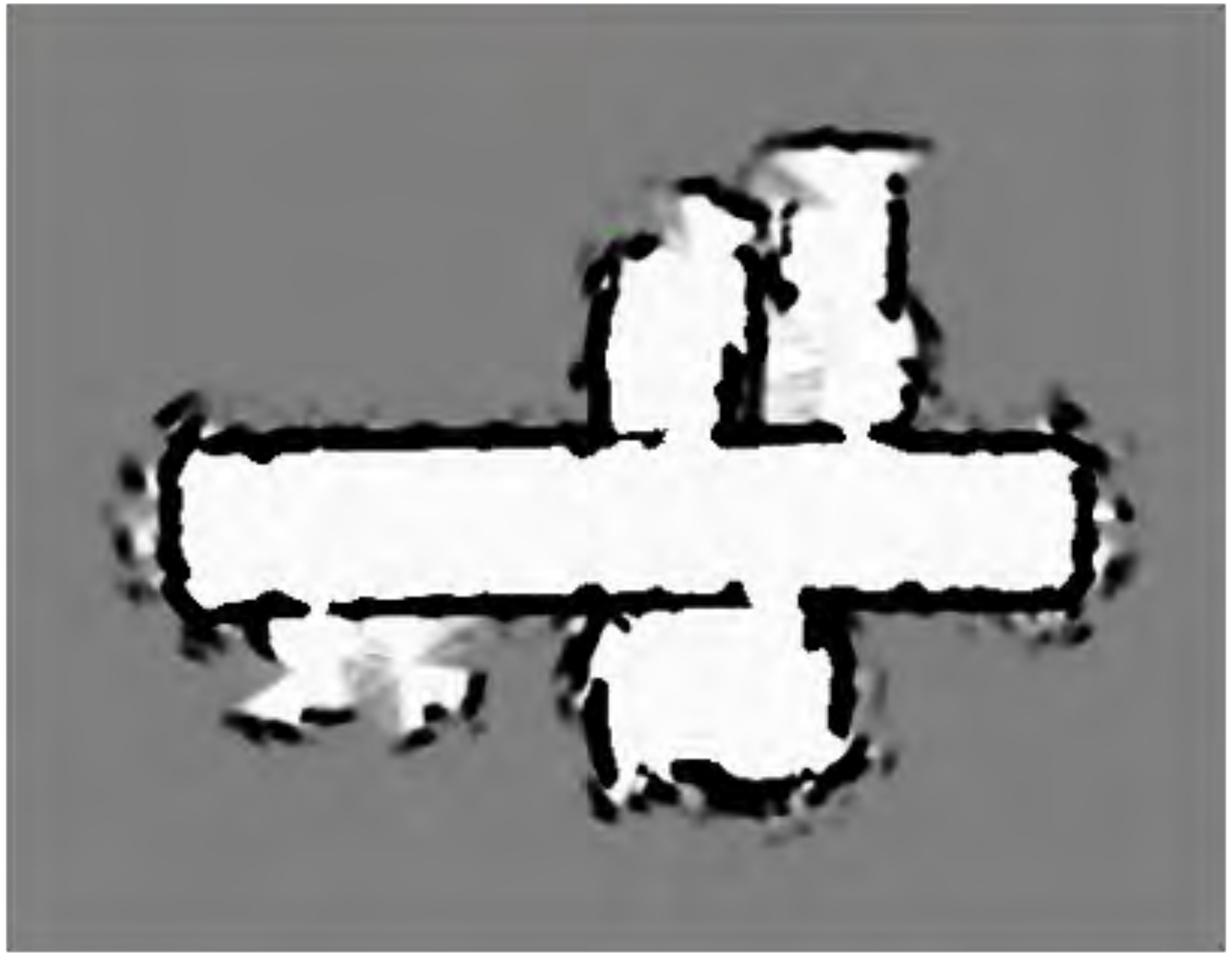
Occupancy Value Depending on the Measured Distance



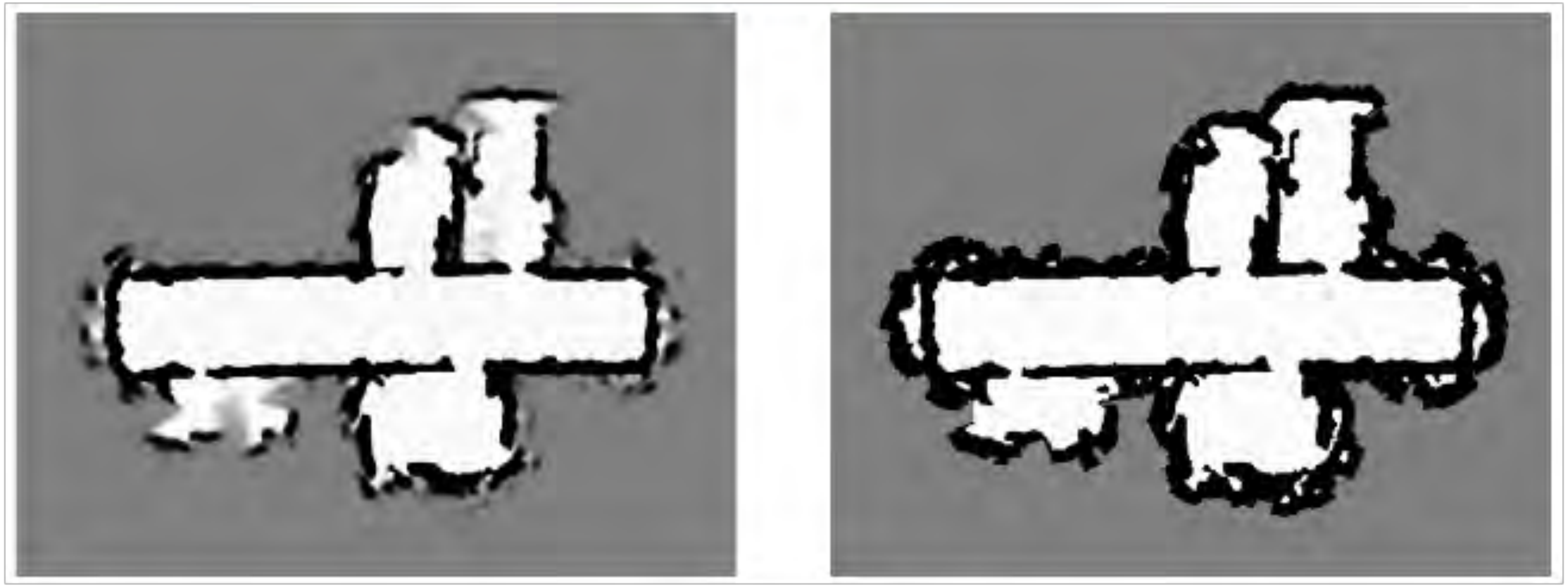
Incremental Updating of Occupancy Grids (Example)



Resulting Map Obtained with 24 Sonar Range Sensors

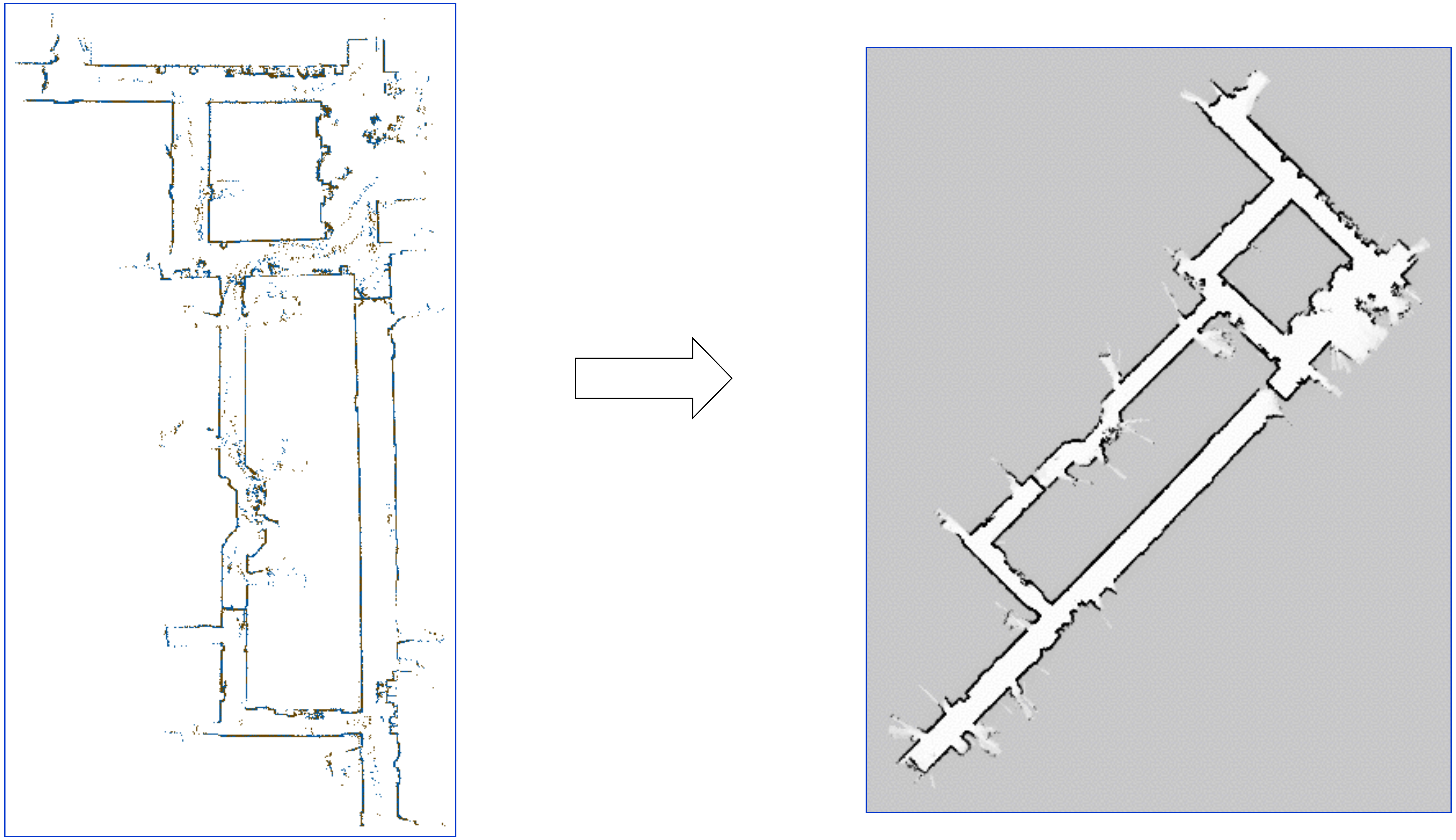


Resulting Occupancy and Maximum Likelihood Map

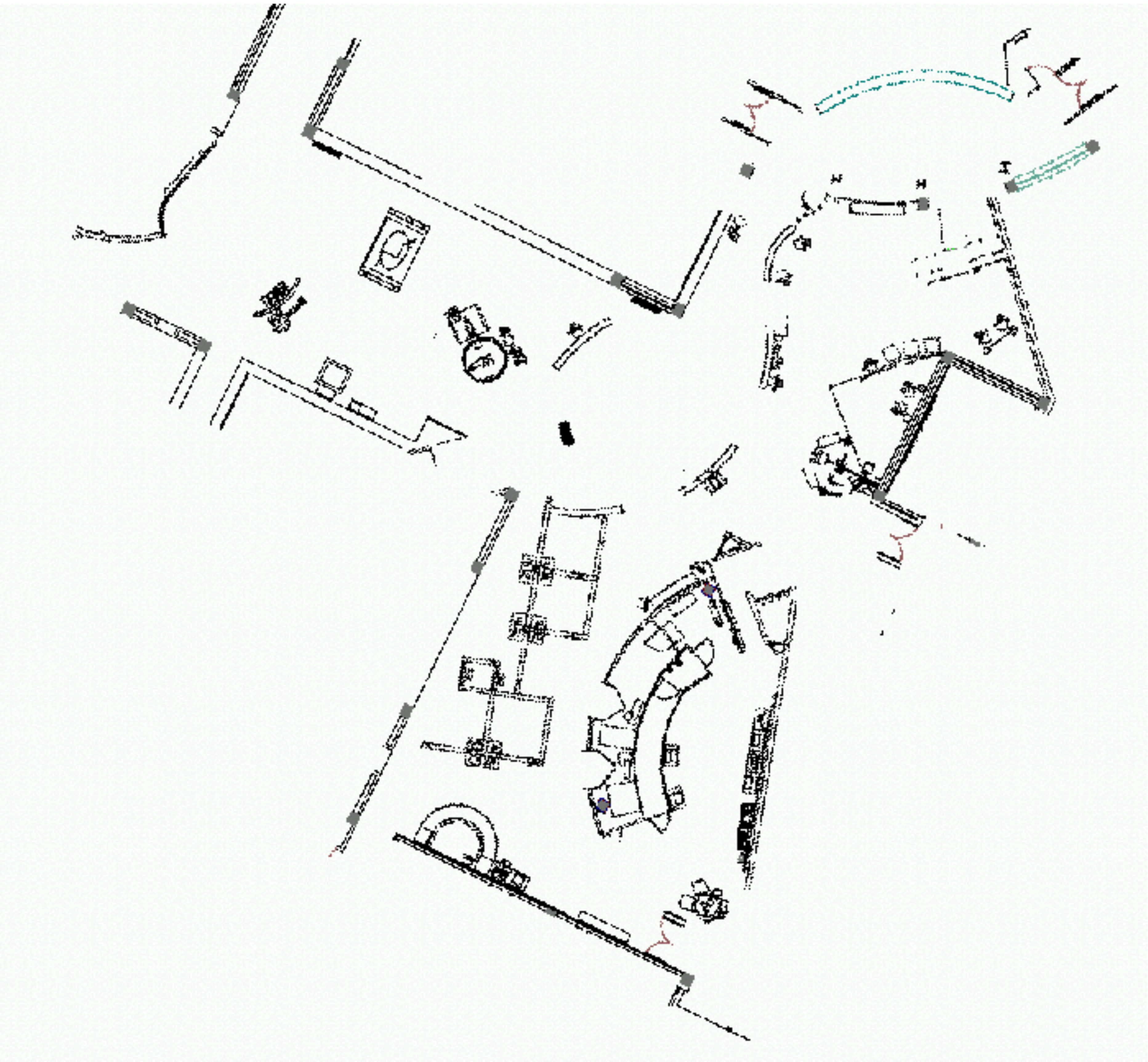


The maximum likelihood map is obtained by rounding the probability for each cell to 0 or 1.

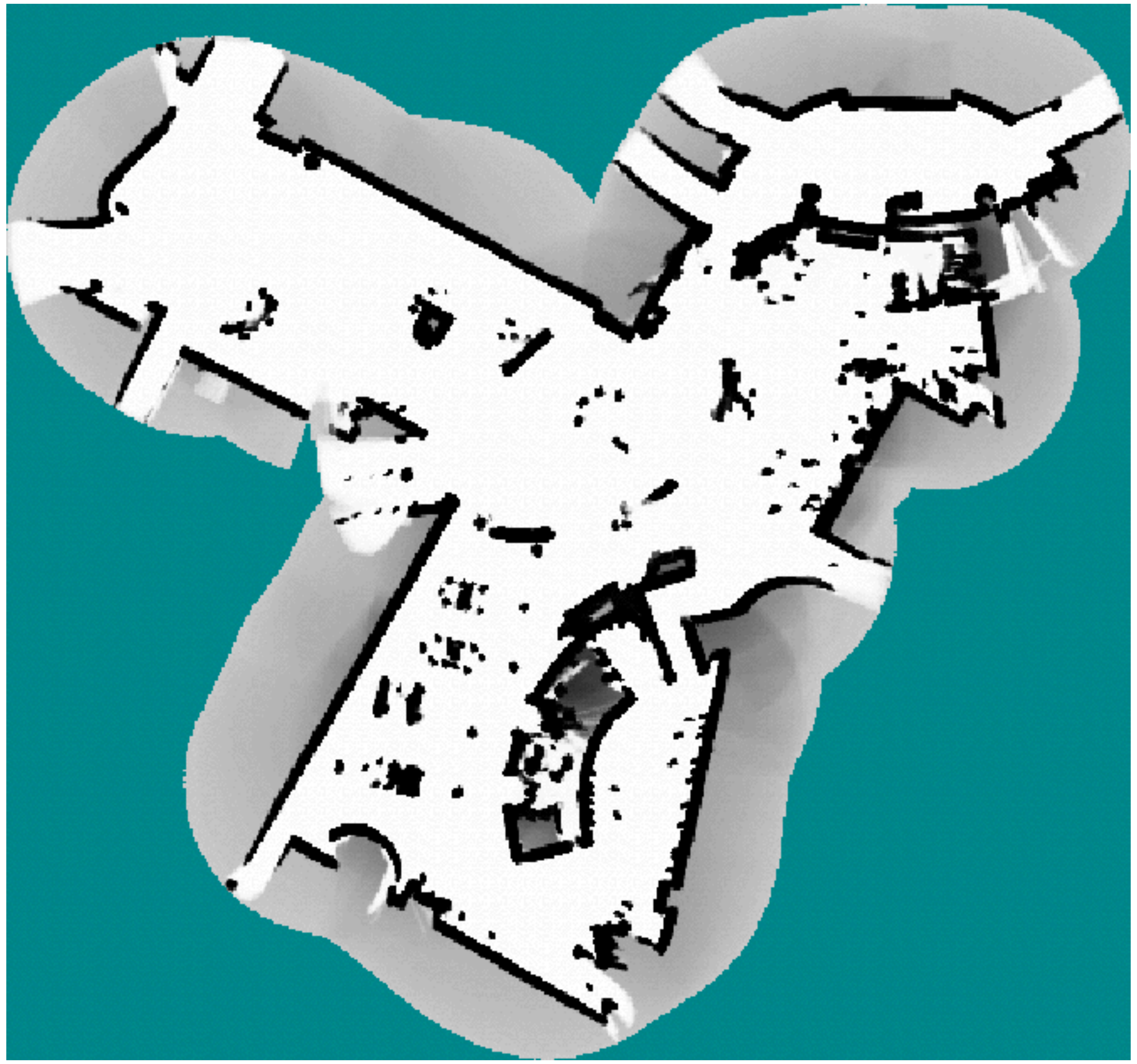
Occupancy Grids: From scans to maps



Tech Museum, San Jose

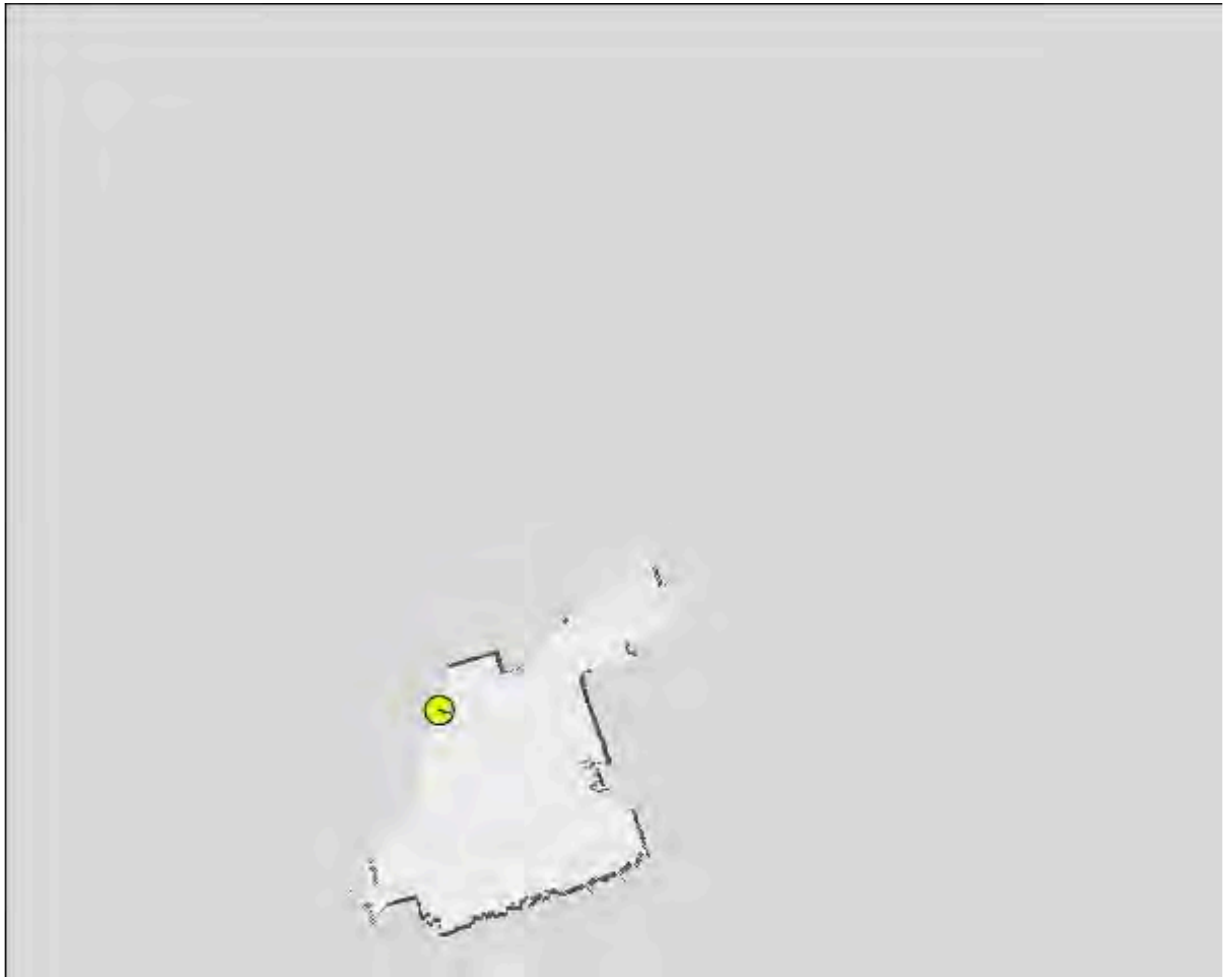


CAD map



occupancy grid map

Uni Freiburg Building 106



Occupancy Grid Map Summary

- Occupancy grid maps discretize the space into independent cells
- Each cell is a binary random variable estimating if the cell is occupied
- Static state binary Bayes filter per cell
- Mapping with known poses is easy
- Log odds model is fast to compute
- No need for predefined features

OctoMap

A Probabilistic, Flexible, and Compact 3D Map Representation for Robotic Systems

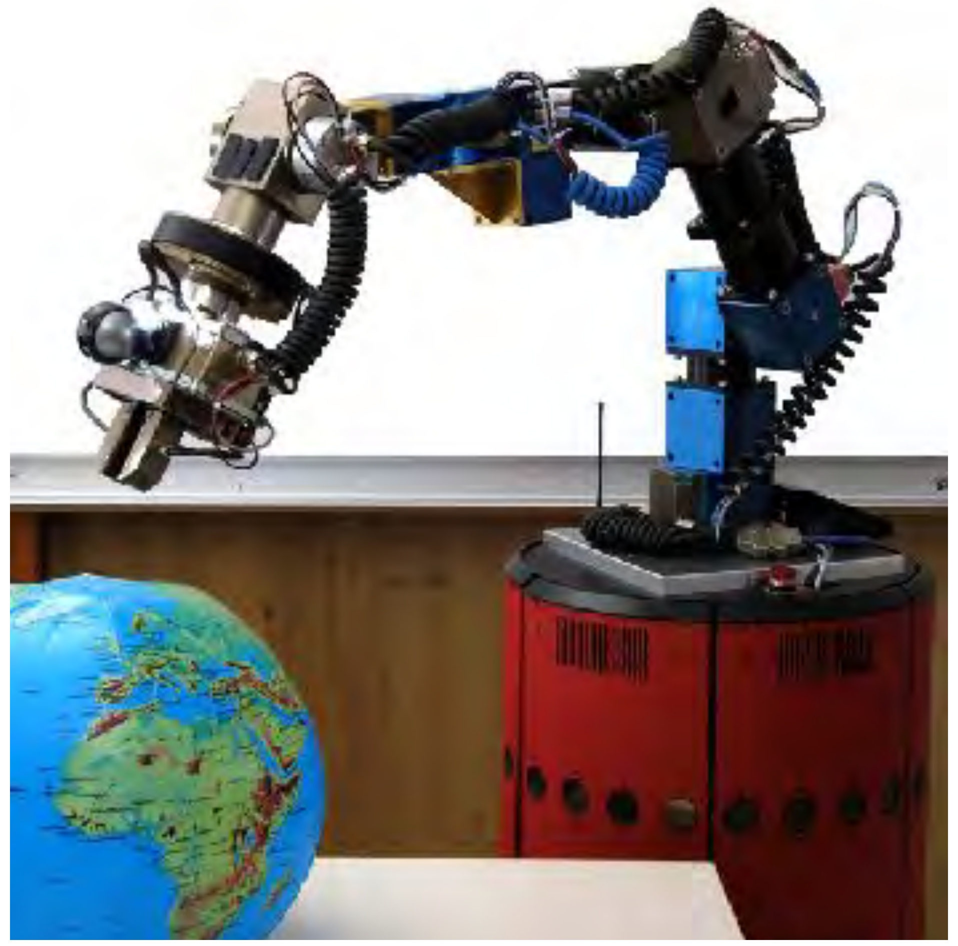
K.M. Wurm, *A. Hornung*,

M. Bennewitz, C. Stachniss, W. Burgard

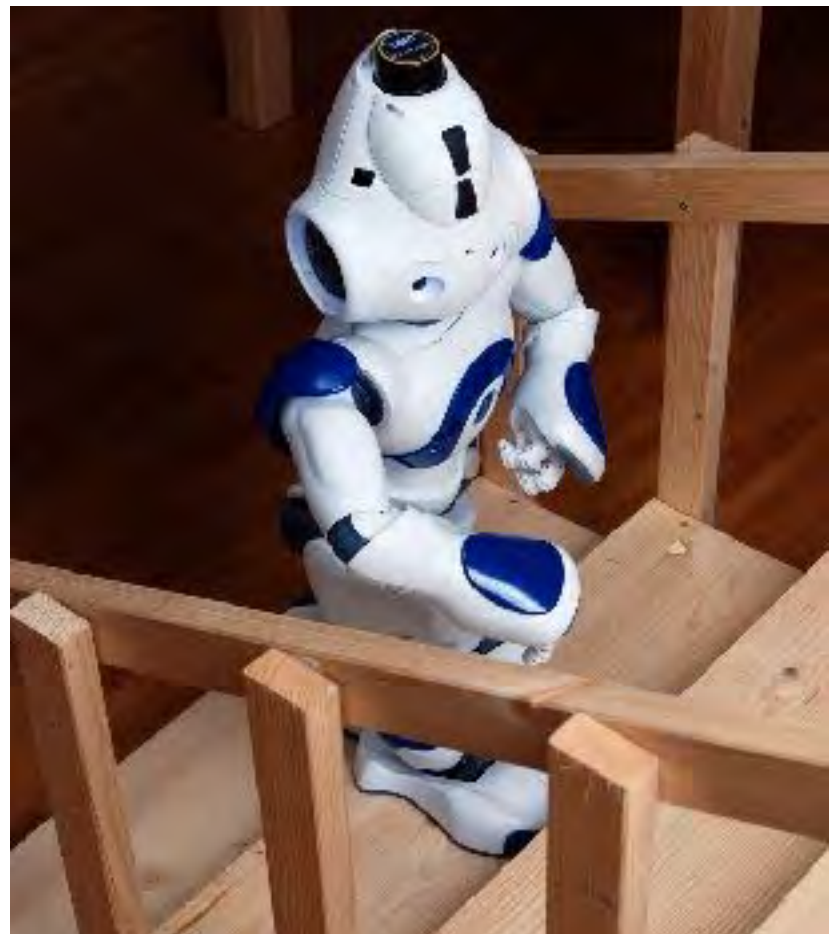
University of Freiburg, Germany

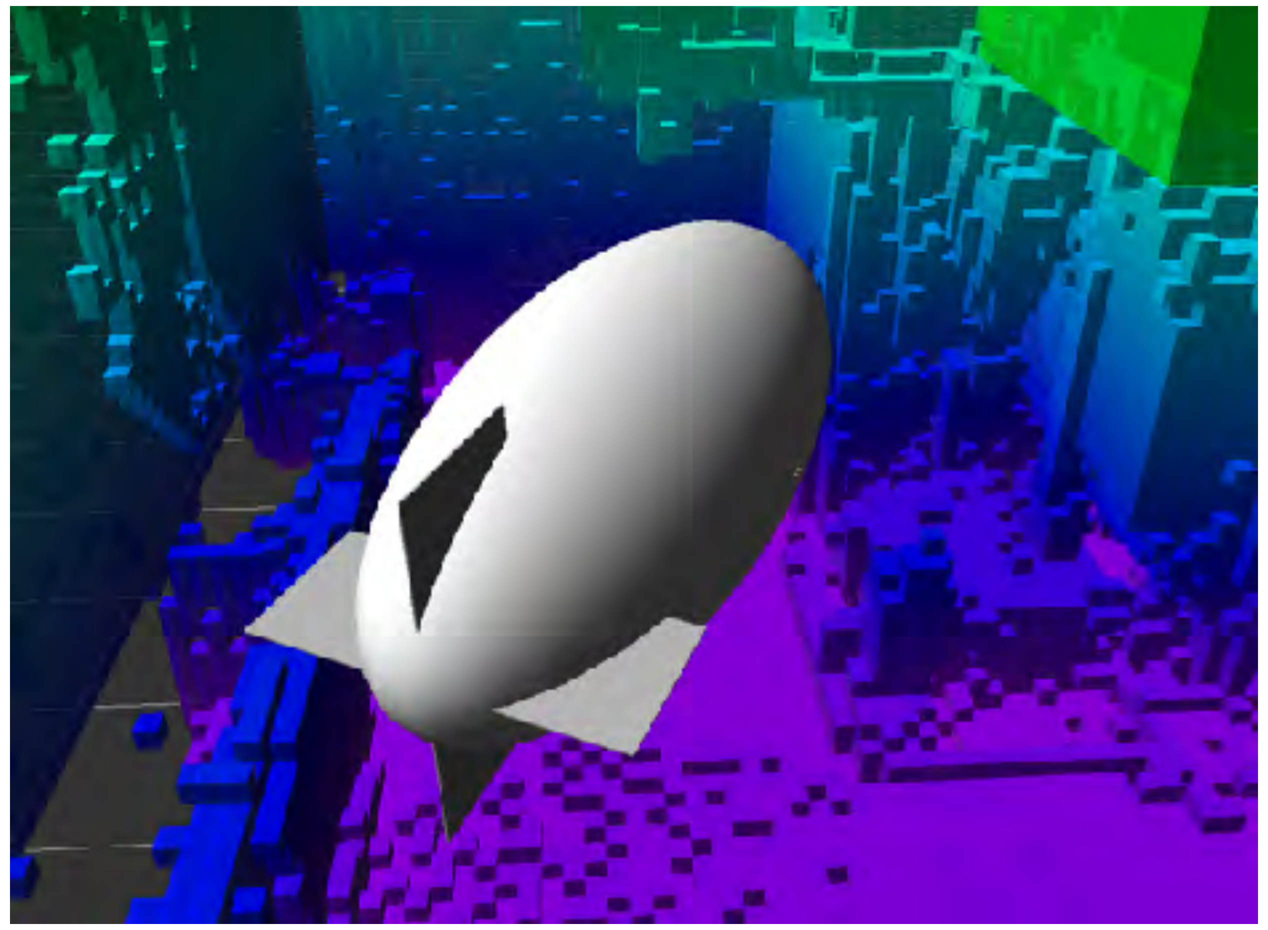
http://octomap.sf.net

Robots in 3D Environments


Mobile manipulation


Outdoor navigation


Humanoid robots

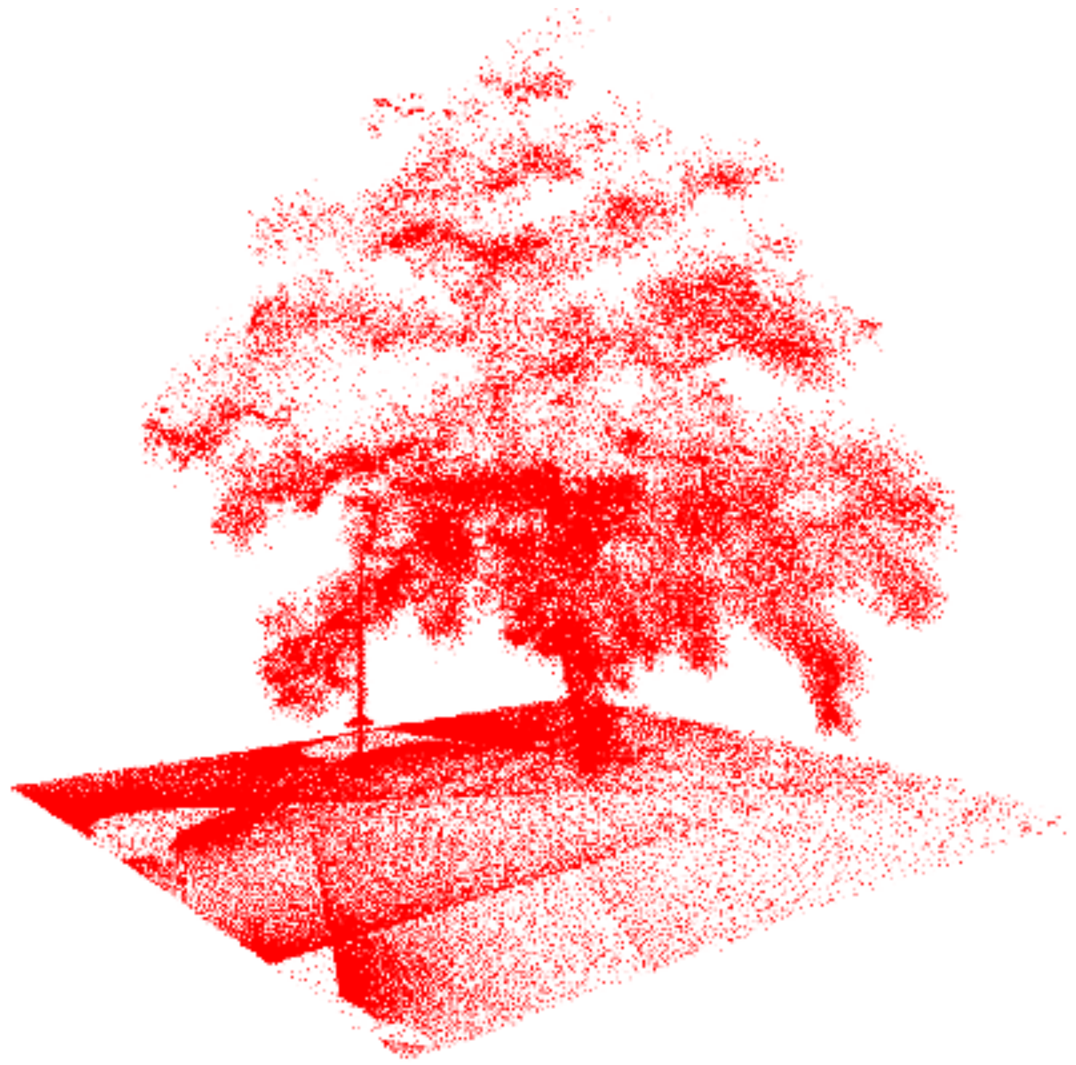

Flying robots

3D Map Requirements

- Full 3D Model
  - Volumetric representation
  - Free-space
  - Unknown areas (e.g. for exploration)
- Can be updated
  - Probabilistic model
    (sensor noise, changes in the environment)
  - Update of previously recorded maps
- Flexible
  - Map is dynamically expanded
  - Multi-resolution map queries
- Compact
  - Memory efficient
  - Map files for storage and exchange

Map Representations

Pointclouds



- Pro:
  - No discretization of data
  - Mapped area not limited
- Contra:
  - Unbounded memory usage
  - No direct representation of free or unknown space

Map Representations

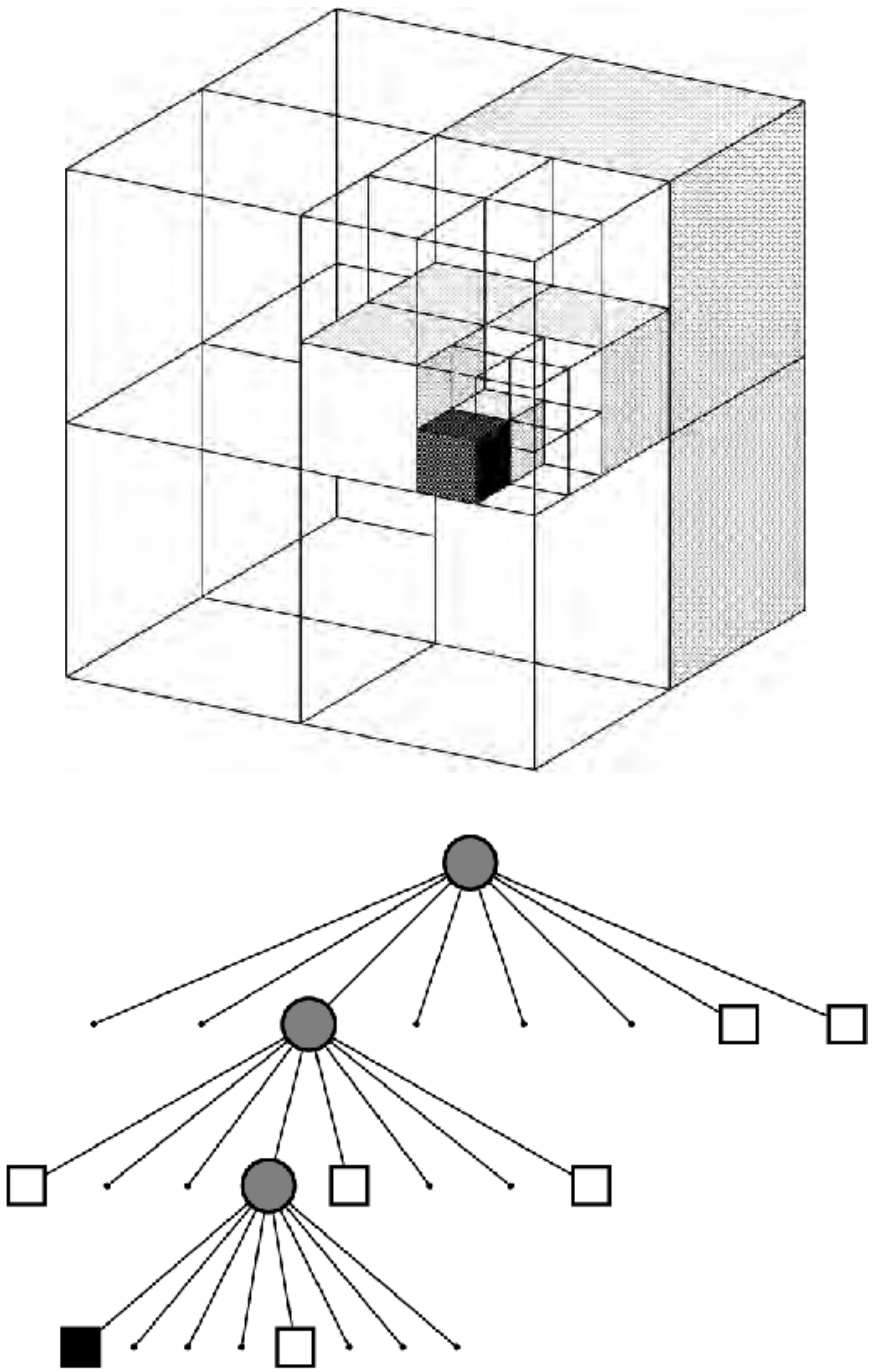
3D voxel grids

- Pro:
  - Probabilistic update
  - Constant access time
- Contra:
  - Memory requirement
    - Extent of map has to be known
    - Complete map is allocated in memory

Map Representations

Octrees

- Tree-based data structure
- Recursive subdivision of space into octants
- Volumes allocated as needed
- Multi-resolution



Map Representations

Octrees

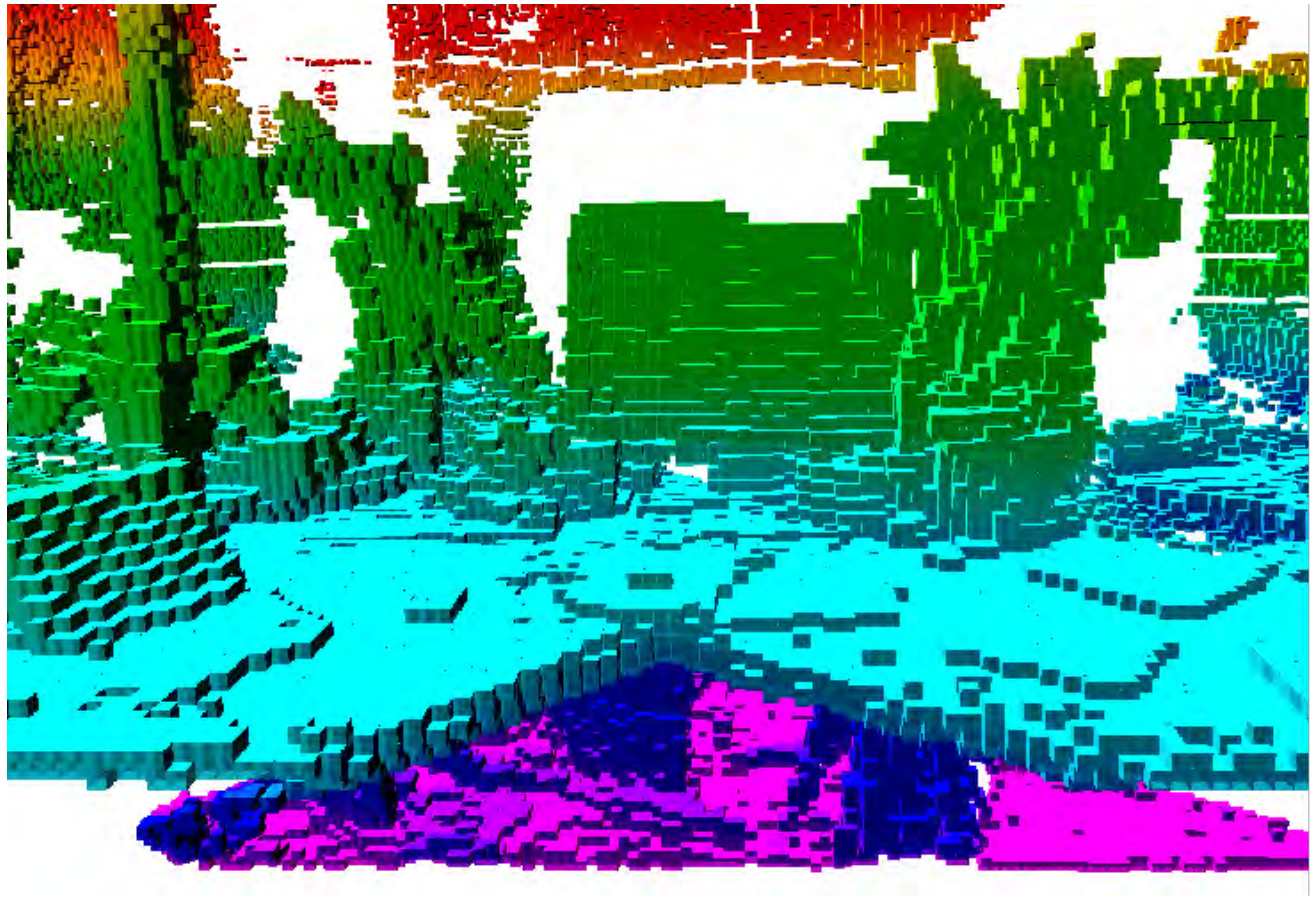
- Pro:
  - Full 3D model
  - Probabilistic
  - Flexible, multi-resolution
  - Memory efficient

- Contra:
  - Implementation can be tricky (memory, update, map files, …)

- Open source implementation as C++ library available at http://octomap.sf.net

Examples

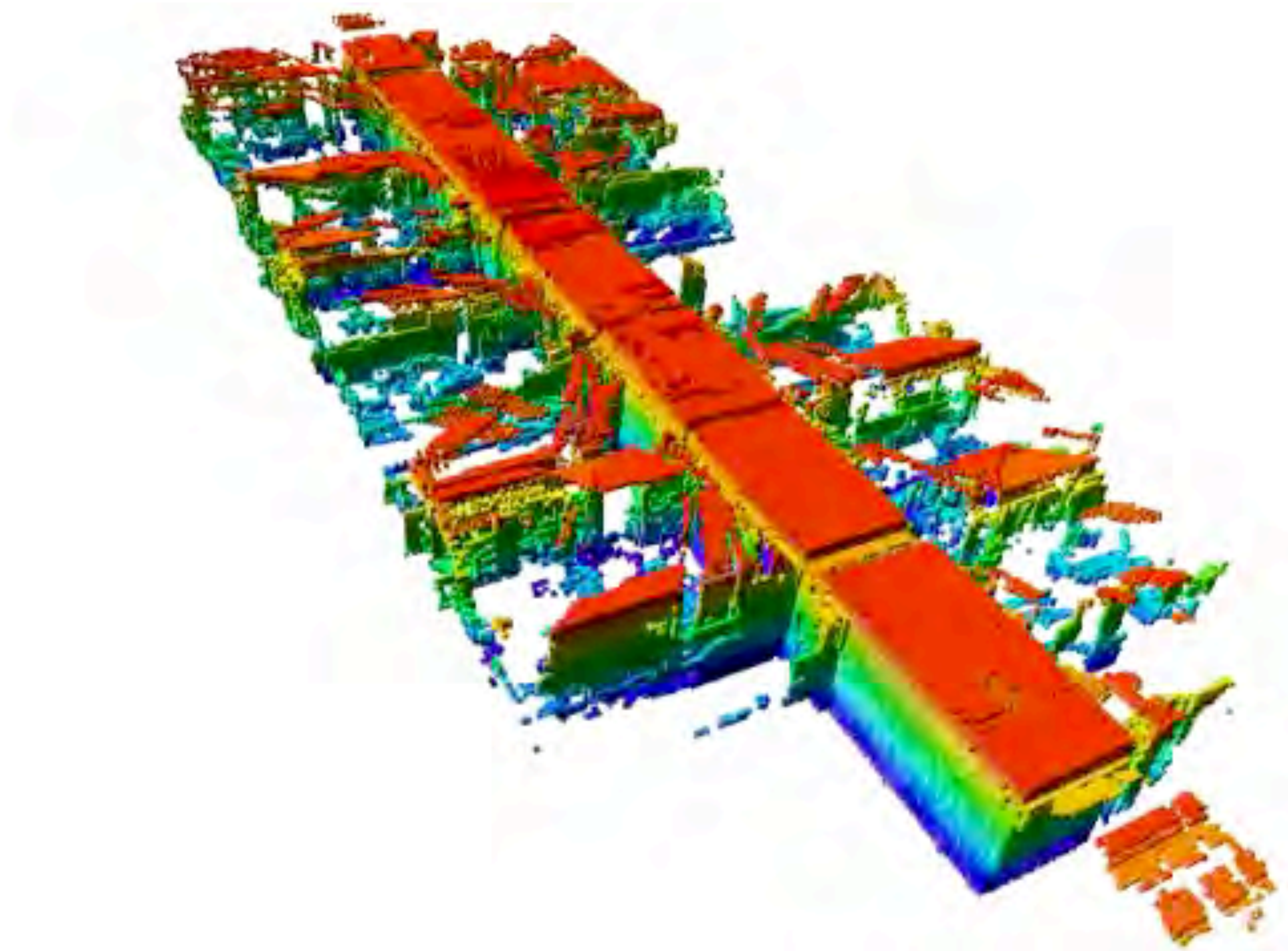
- Cluttered office environment



Map resolution: 2 cm

Examples: Office Building

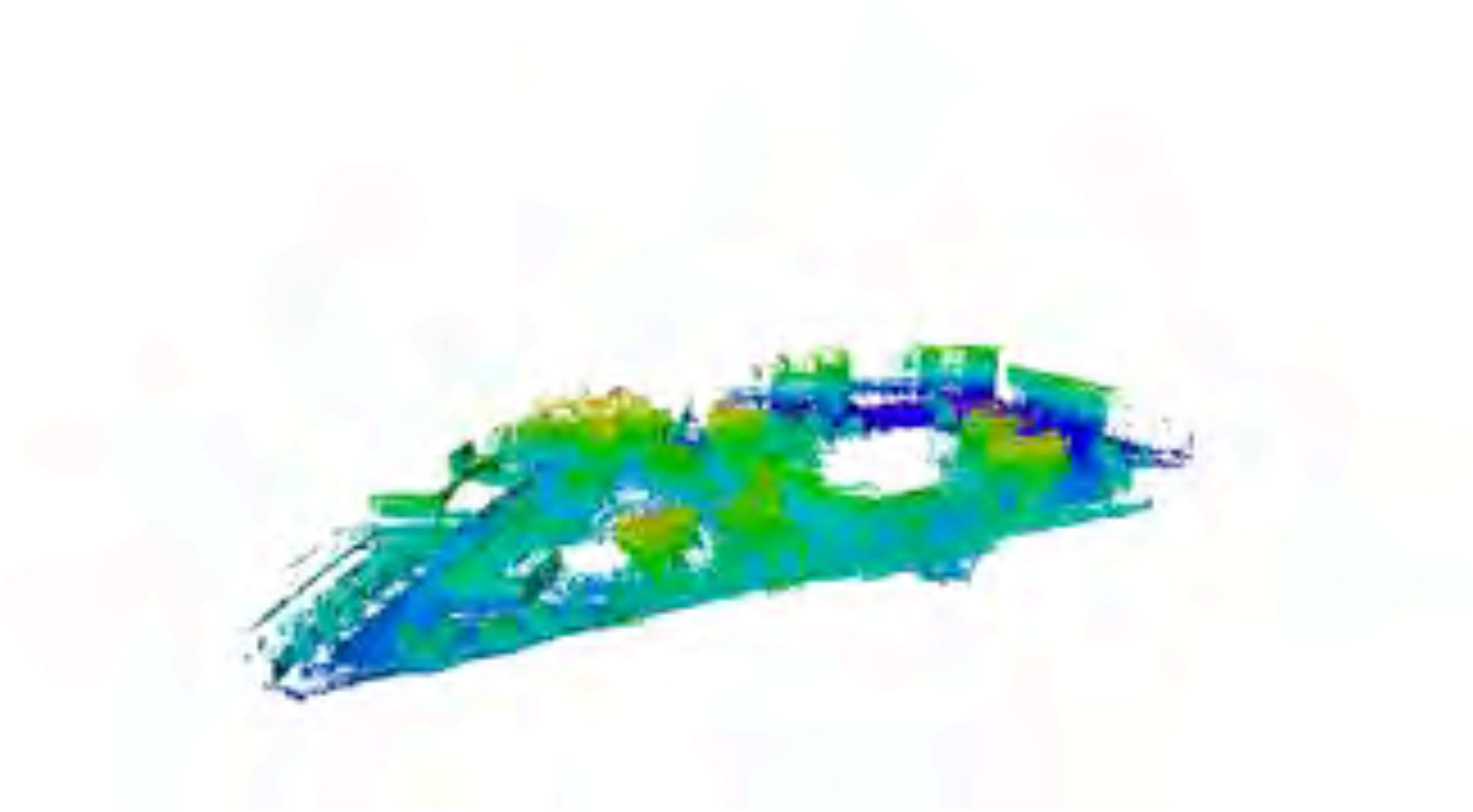
- Freiburg, building 079



Examples: Large Outdoor Areas

- Freiburg computer science campus

  (292 x 167 x 28 m^3, 20 cm resolution)



Slide borrowed from Dieter Fox

Examples: Tabletop

Frontier-based Exploration:

Frontier-based exploration is the process of repeatedly detecting frontiers and moving towards them, until there are no more frontiers and therefore no more unknown regions.

What are frontiers?

Frontier cells define the border between known and unknown space.

Next Lecture:
SLAM