Intelligent Robotic Systems

CS 485

Jana Kosecka, 4444 Research II
kosecka@gmu.edu , 3-1876
Office hours Wed 2-3pm

Logistics

- **Grading:** Homeworks 40%, Project 30% Exam: 30%
- **Prerequisites:** basic statistical concepts, geometry, linear algebra, calculus, CS 480
- **Course web page** cs.gmu.edu/~kosecka/cs485/

- Homeworks about every 2 weeks, Midterm, Final Project
- Choose from the list of projects, suggest your own
- Implement one of the covered methods on robot/robot simulator, come up with new ideas of robotics tasks, implement techniques on real robot
- Write a report and prepare the final presentation
- I would encourage use of open source tools

Recommended Text

- R. Siegwart and I. Nourbakhsh: **Introduction to Autonomous Mobile Robots**, MIT Press, 2004
- [1] S. LaValle: **Planning Algorithms**, Cambridge Press,
- http://planning.cs.uiuc.edu/
- [2] S. Thrun, W. Burghart, D. Fox: **Probabilistic Robotics**
- http://robots.stanford.edu/probabilistic-robotics/
- [4] S. Russell and P. Norvig: **Artificial Intelligence: A Modern Approach**
- [5] R. Sutton and A. G. Barto: **Introduction to Reinforcement Learning** (on-line materials see course www)

Course Logistics

- Required Software MATLAB (with Image Processing toolbox), Octave (open source Matlab like language)
- Robot simulators, real robots

- Availability of robotics platforms
- Pioneers with range sensors, cameras
- Humanoid – Small soccer league
- Simulators
- List of resources http://www.mobilerobots.org//

Focus of the course

- General introduction and techniques
- Provide overview of the approaches in mobile robotics
- Motion Planning
- Perception, control, localization
- Probabilistic robotics

- Hand of experience with simulation, programming and real robots

- Current trends and areas of robotic technologies

+ little history

Applications - Robots in manufacturing/material handling

Manhattan project (1942) – handling and processing of radioactive materials – Telemanipulation

Manufacturing
- storage, transport delivery
- table top tasks, material sorting, part feeding – conveyor belt
- microelectronics, packaging
- harbor transportation
- construction (automatic cranes)

Suitable for hard repetitive tasks – heavy handling or fine positioning
Successful in restricted environments, limited sensing is sufficient – limited autonomy

Autonomous Robotic Systems
AGV's - automated guided vehicles
AUV's - automated unmanned vehicles

Applications - Space Robotics

50-ties US space program, exploration of planets, collecting samples
Astronauts bulky space suits – difficult
NASA, JPL, DARPA – sponsoring agencies
Space programs, military application – surveillance, assistance

Planetary Rovers – initially controlled by humans
- large time delays,
- poor communication connections

Need for (semi) – autonomy

Teleoperation – Mars Rover

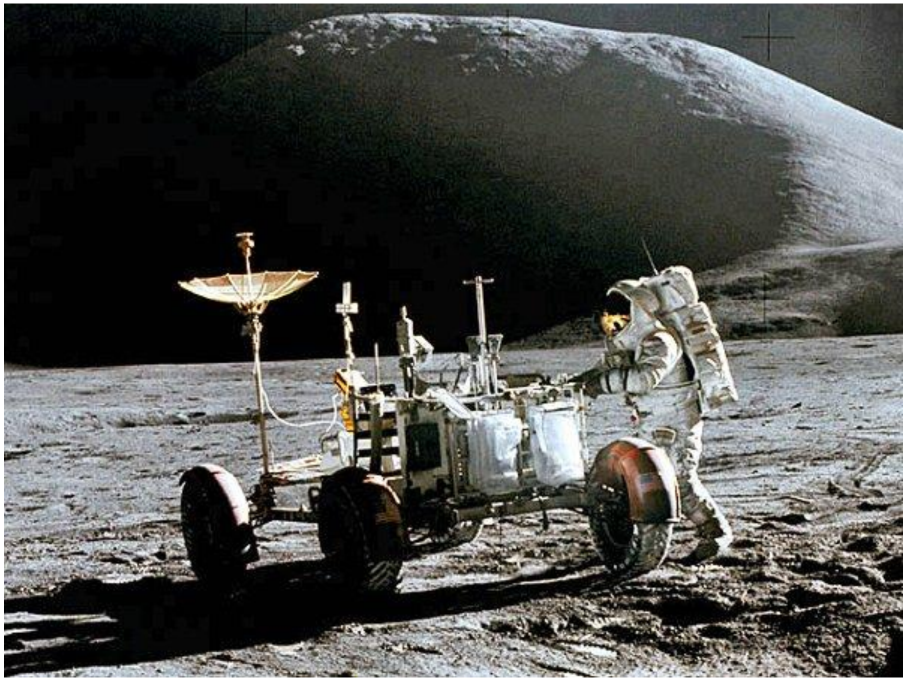
Human operator controls the robot
Local site – human views the sensory data, sends the commands
Remote site – sensors acquire the information

Example 1:Building Virtual Models of Mars

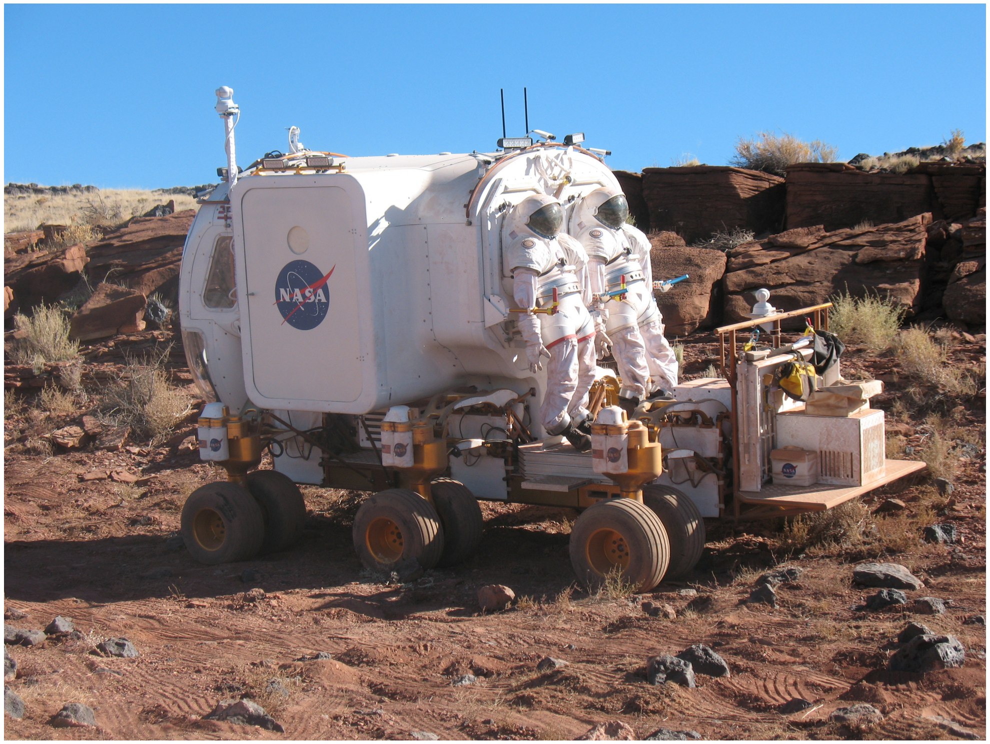
Example of stereo pipeline, from raw data, preprocessing, meshes, texture maps

See http://schwehr.org/photoRealVR/example.html

Appollo



Lunar Rovers



Current NASA Prototype

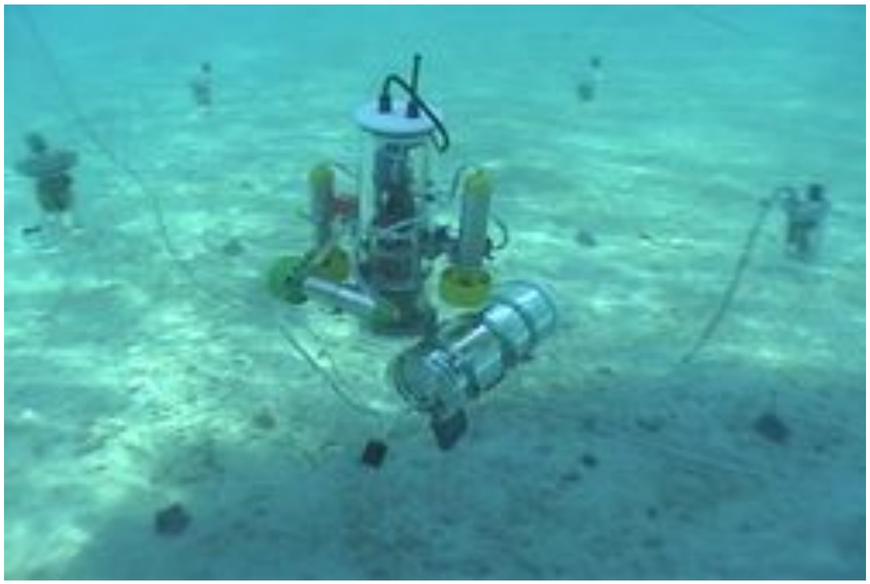
Applications: Navigation in difficult terrain/ harsh conditions

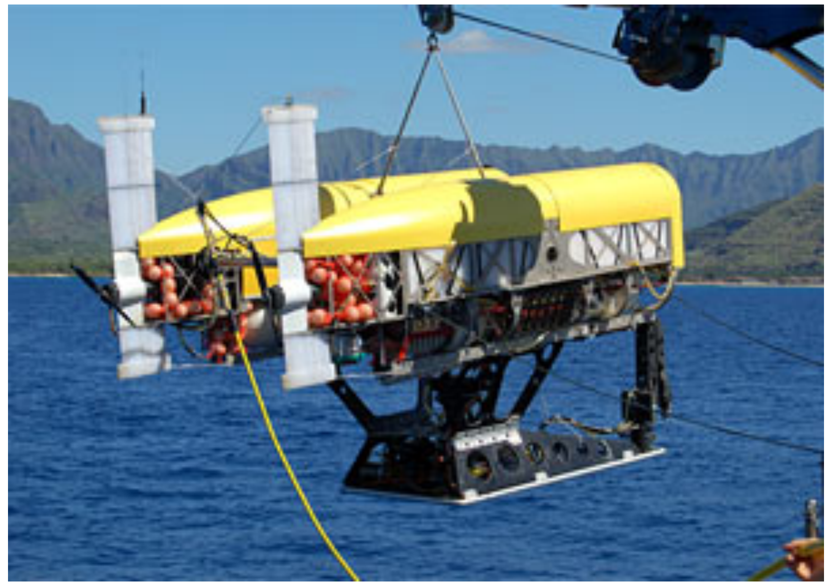
- Antarctica – search for samples of meteorites
- Volcanos – analyze gas samples from volcanos

Applications: Underwater robotics

- Sensor network

- Remotely Operated robot for ocean exploration





Robots in the service of humans

- Robotic surgery - DaVinci robotic surgery robot – human assisted
- http://www.intuitivesurgical.com/products/da_vinci_video_overview.aspx
- Robotics in rehabilitation surgery (Hocomo Inc)



- **Mobile Robots**

- courier in buildings and hospitals, vacuum cleaners,

Variety of domains and tasks

Games and Entertainment

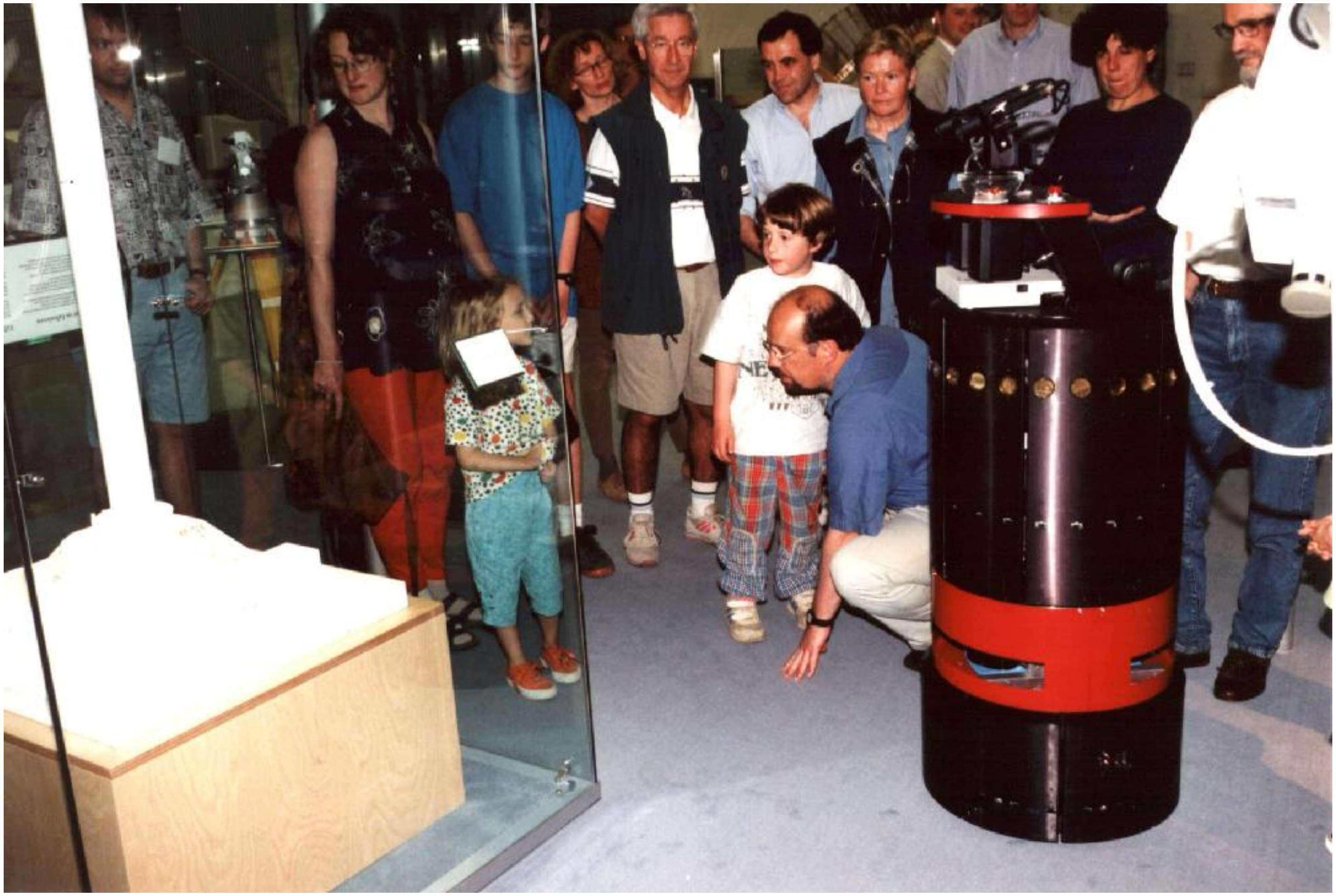
Furbies

Aibos Latter & Macaron

Aibo soccer league - RoboCup

Rhino – First Museum Tour giving robot

University of Bonn ('96)

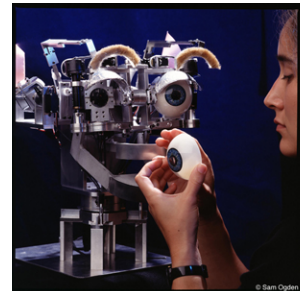


Toy Robot Aibo from Sony

- Size
  - length about 25
- Sensors
  - color camera
  - stereo microphone

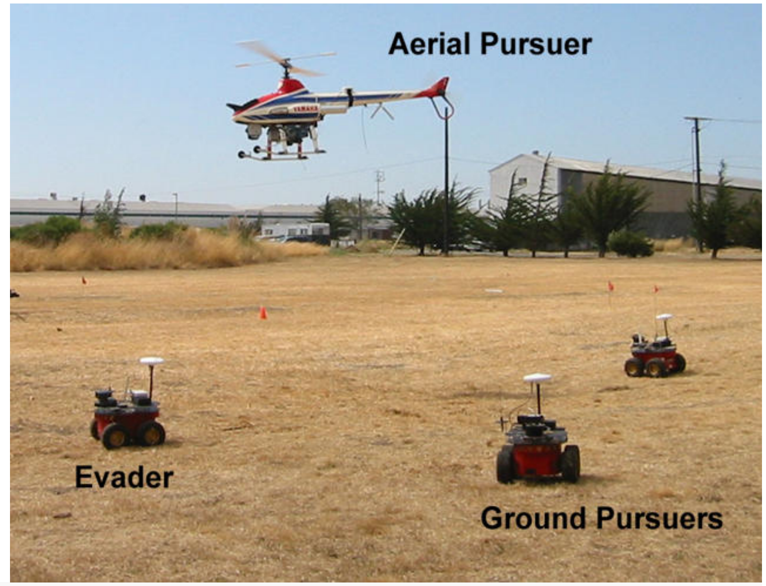
Humanoid Robots

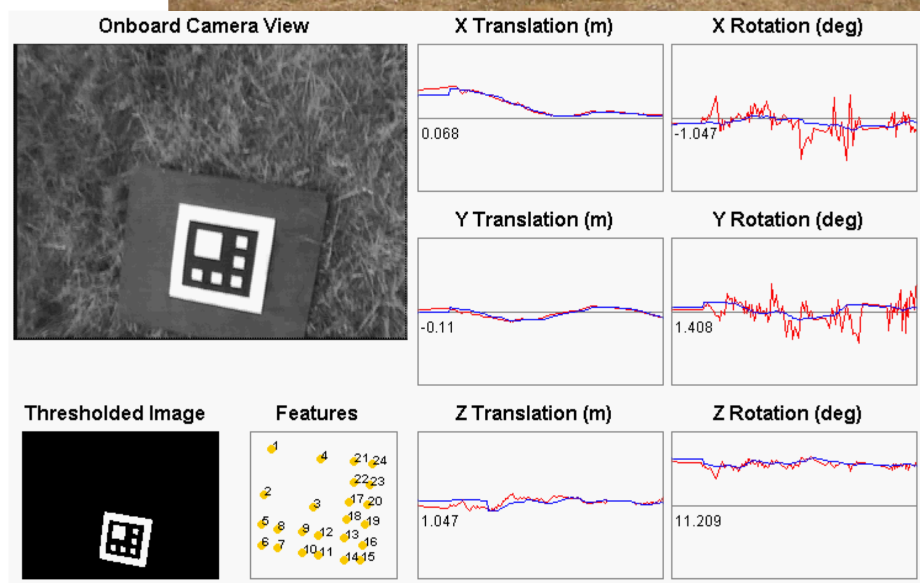
by HONDA



MIT Cog Project

APPLICATIONS – **Unmanned Aerial Vehicles (UAVs)**





Rate: 10Hz
Accuracy: 5cm, 4º

Berkeley Aerial Robot (BEAR) Project

Robotic Navigation

- Stanford Stanley Grand Challenge
- Outdoors unstructured env., single vehicle

- Urban Challenge
- Outdoors structured env., mixed traffic, traffic rules

Intelligent Robotic System

- Mechanical System with some degree of autonomy

- Three Basic Components of the Intelligent Robotic System
- SENSE – process information from the sensors
- PLAN – compute the right commands/directives
- ACT – produces actuator commands

- Different organization of these functionalities gives rise to different robot architectures

Robotics and AI

Knowledge representation
- how to represent objects, humans, environments
- symbol grounding problem

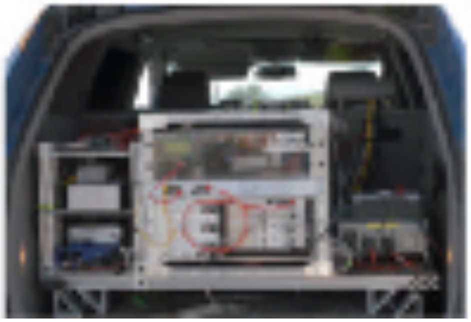
Computer Vision, Pattern Recognition
- study of perception
- recognition, vision and motion, segmentation and grouping
  representation

Natural Language Processing
- provides better interfaces, symbol grounding problem

Planning and Decision Making
How to make optimal decision, actions give the current knowledge of the state, currently available actions

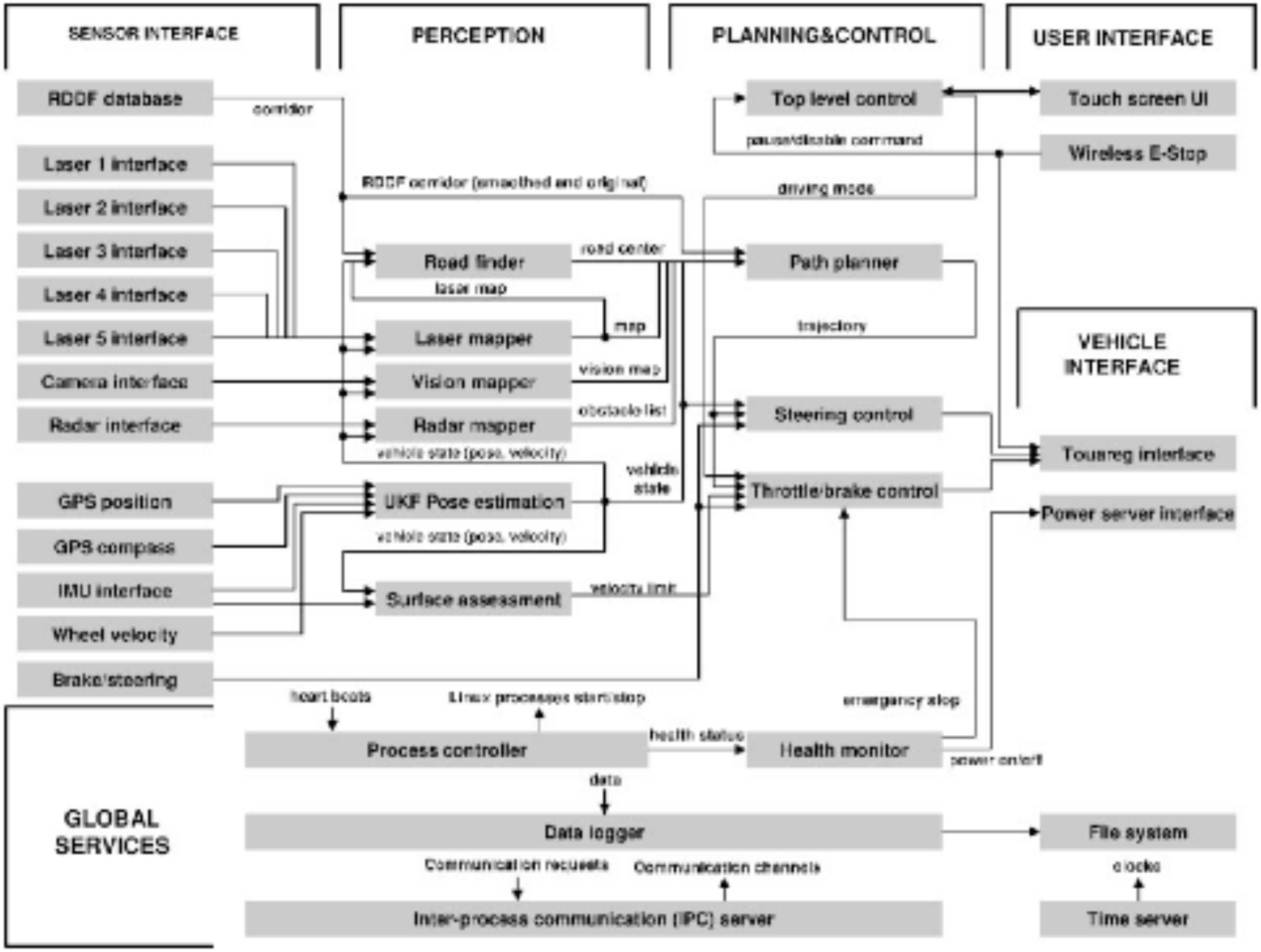
Robot Components (Stanley)

- Sensors
- Actuators-Effectors
- Locomotion System
- Computer system – Architectures – (the brain)

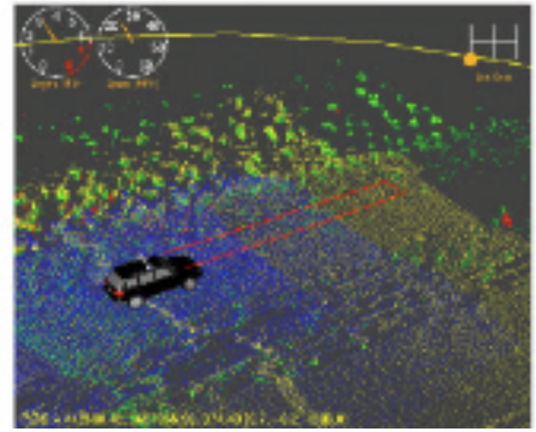




- Lasers, camera, radar, GPS, compass, antenna, IMU,
- Steer by wire system
- Rack of PC's with Ethernet for processing information from sensors

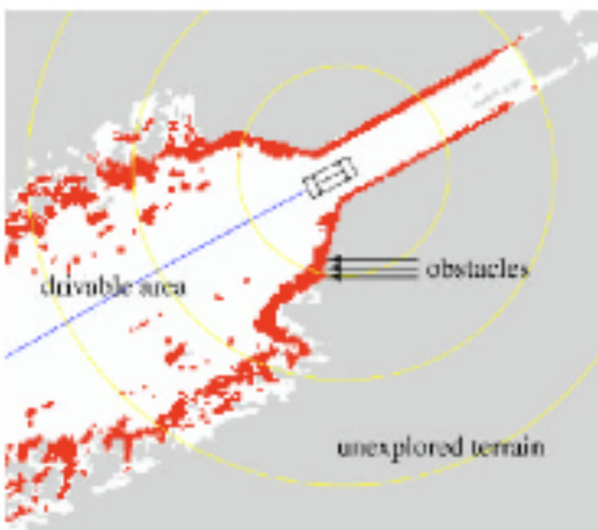
Stanley Software System

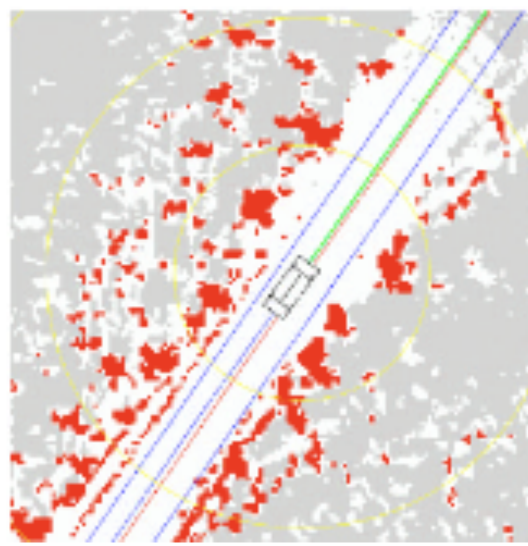


- Terrain mapping using lasers



- Determining obstacle course





Robot Components

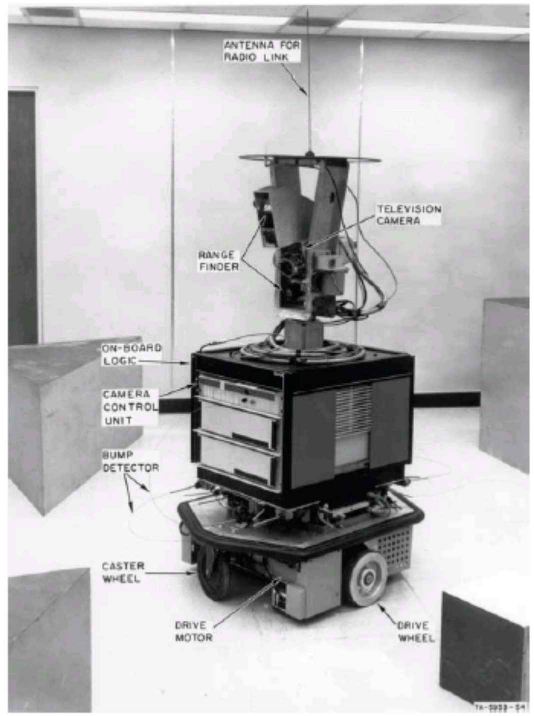
Sensors:

- State is determined by measuring some physical quantity – voltage, current, distance
- Environment is often dynamic and unknown, robot has only uncertain knowledge of the environment due to the limited and noisy sensing capabilities
- External state - state of environment, temperature, presence of obstacles, people
- Internal state – state of the robot, position, orientation, force, battery charge, (happy, sad)

Actuators

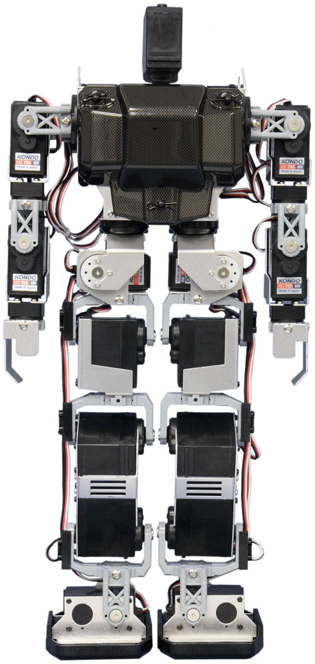
- Robot can change it's state and the state of the world by means of actuators
- Actuators for locomotion
- Actuators for manipulation
- Convert software commands into physical actions
- (hydraulic, electric, pneumatic)
- Domain of mechanical engineering – new actuator designs (weight, flexibility)
- Actuators – have inaccuracies and are often not true to their model – difficult calibration issues, wear and tear, need to adaptive

Shakey the Robot



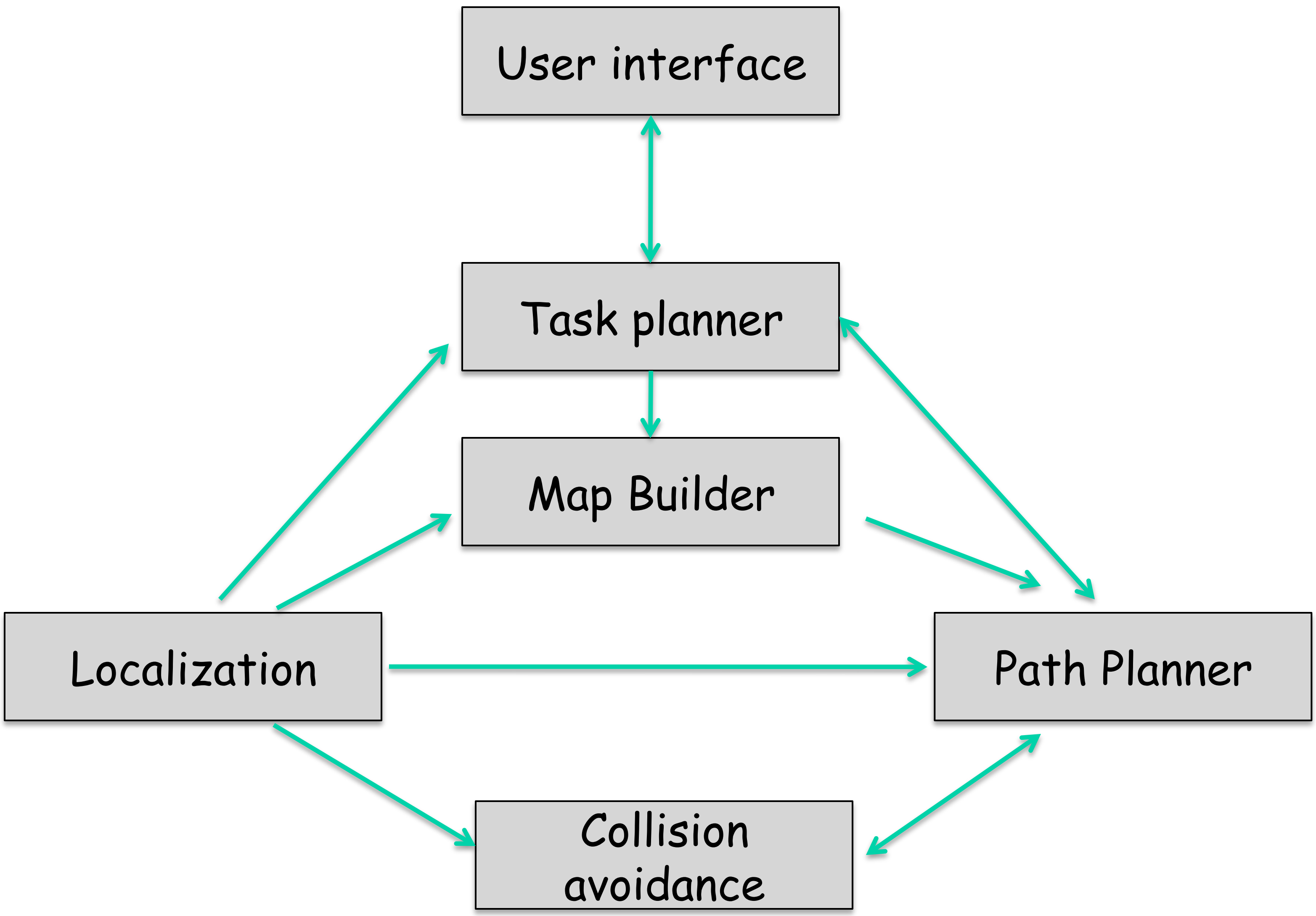
Shortcut to Sh

Robots and GMU



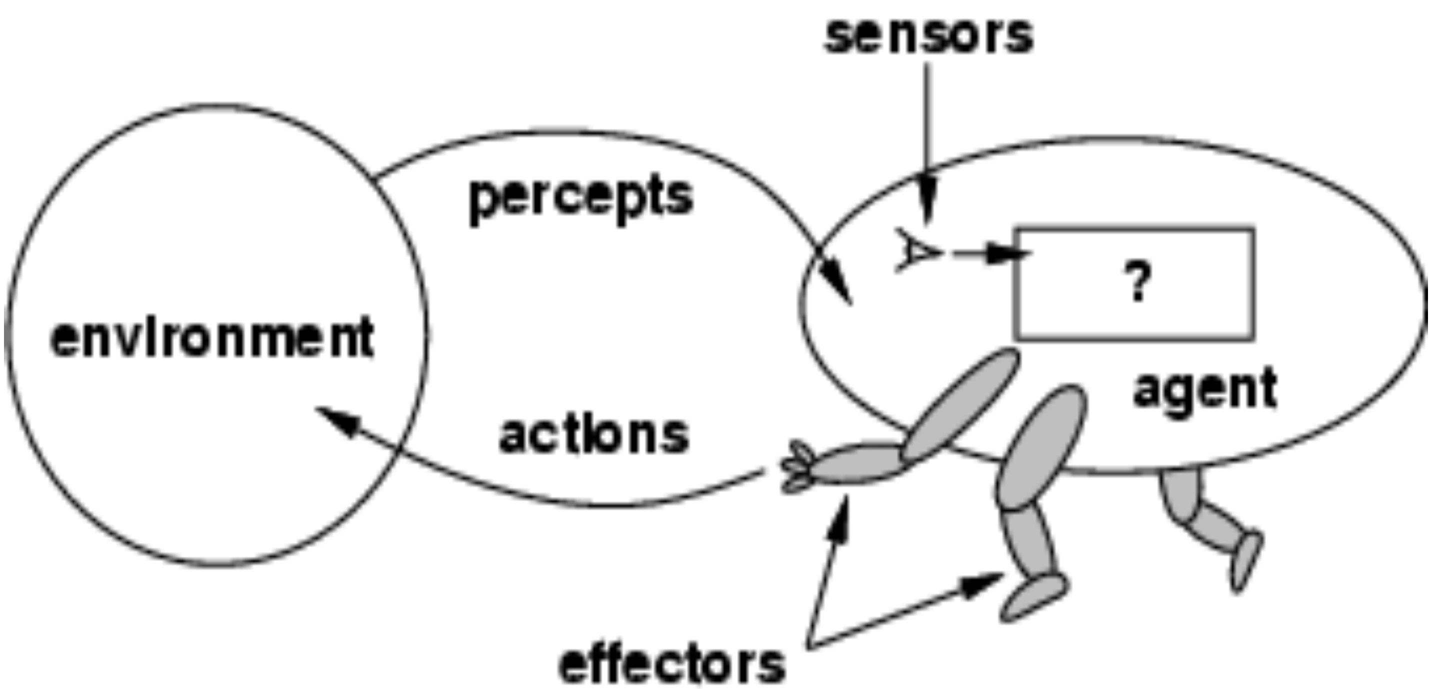
- Marker based motion capture systems
- Haptics phantoms
- EEG

Typical architecture



Agents and Environments

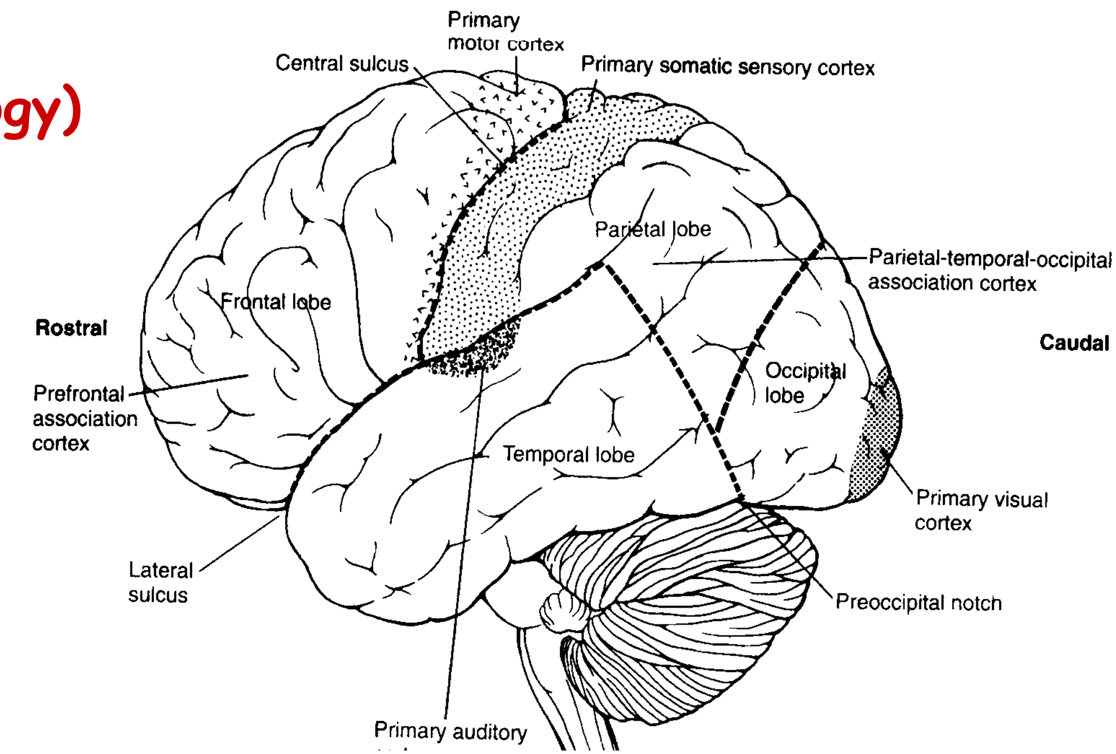
Different Computational paradigms
(Russell & Norvig)



- Computational Aspects/Ingredients
- percepts, actions, goals, environments

How to do the right thing ? (sensory processing, planning, control)

The Brain (analogy)

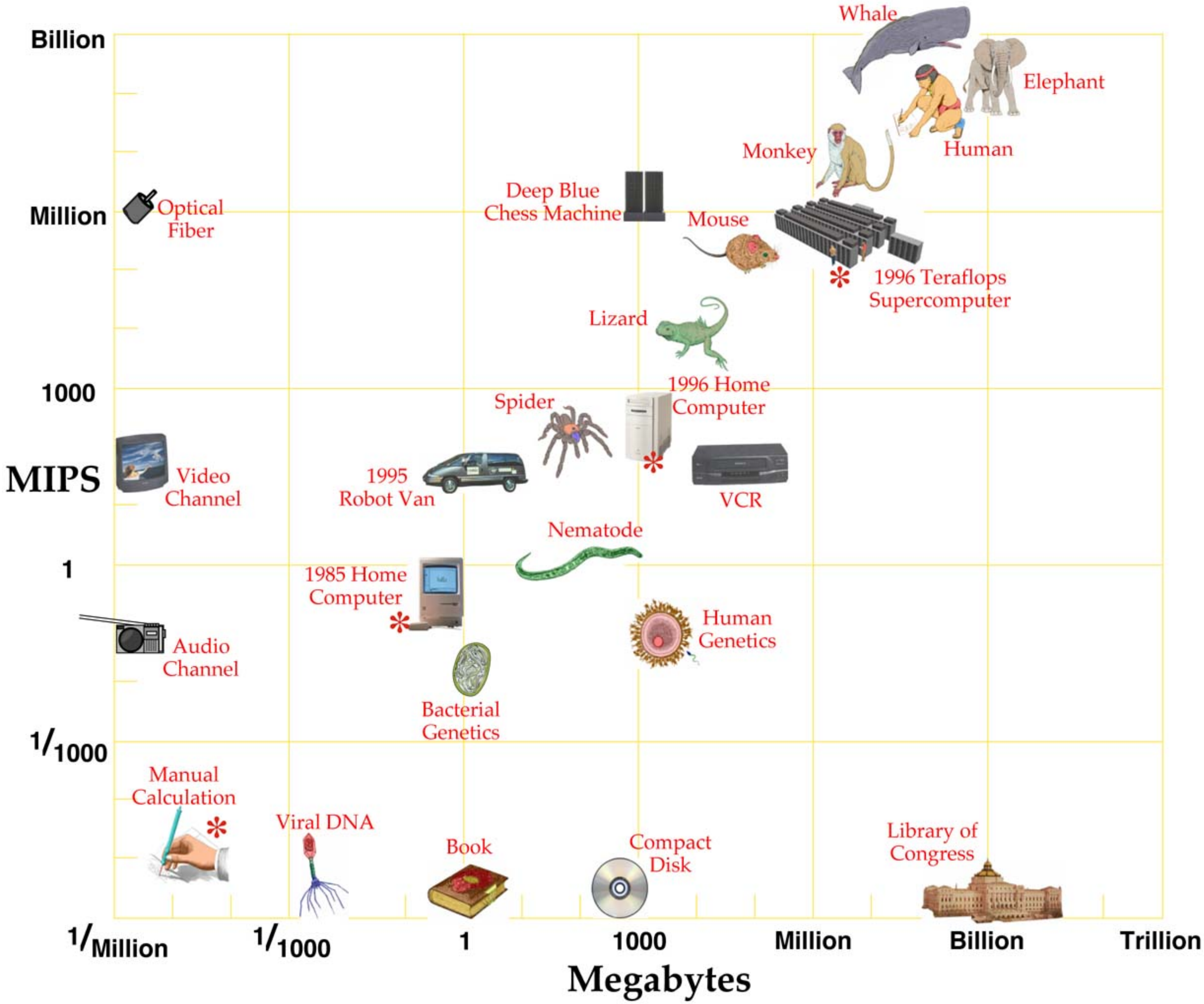


- 100 Billion neurons
- On average, connected to 1 K others
- Neurons are slow. Firing rates < 100 Hz.
- Can be classified into
  - Sensory – vision, somatic, audition, chemical
  - Motor – locomotion, manipulation, speech
  - Central – reasoning and problem solving

Trends in biological and machine evolution
Hans Moravec: Robot

- 1 neuron = 1000 instructions/sec
- 1 synapse = 1 byte of information
- Human brain then processes 10^{14} IPS and has 10^{14} bytes of storage
- In 2000, we have 10^{9} IPS and 10^{9} bytes on a desktop machine
- In 25 years, assuming Moore's law we obtain human level computing power

All Thinks, Great and Small



Overview of the topics

- Kinematics, Kinematic Chains, Mobile Robot kinematics
- Notion of state, sensing state, elementary control
- Potential Field Based Methods, Robot Behaviors

- Robot Perception – Sensors, Visual Perception
-
- Foundations of Probabilistic Robotics
- State estimation and Tracking
- Localization using Particle Filters
- Simultaneous Localization and Mapping

- Configuration Space, Motion Planning
- Dynamic Programming and Markov Decision Processes
- Learning how to act – Reinforcement Learning

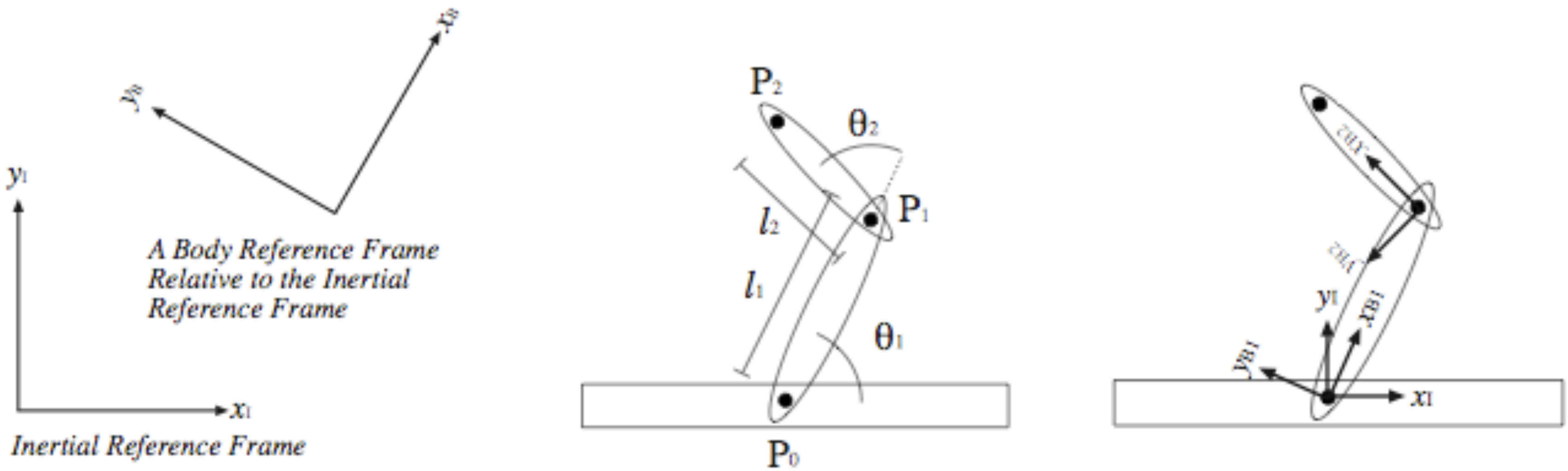
Course Overview – PART I

- Modeling aspects of the robotic system
- Notion of state, state evolution, kinematics
- Systems view suppose vector $\mathbf{x}$ denotes the state of the system, vector $\mathbf{u}$ types of controls/actions the system can carry out we will discuss ways of characterizing the motion of the system

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t)$$

$$\dot{\mathbf{x}(t)} = f(\mathbf{x}(t), \mathbf{u}(t))$$

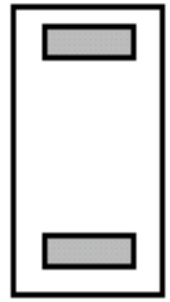
Modeling Geometric transformation



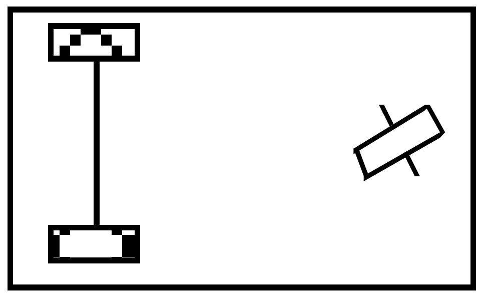
- Modeling Rigid Body Motion
- Modeling Kinematic Chains

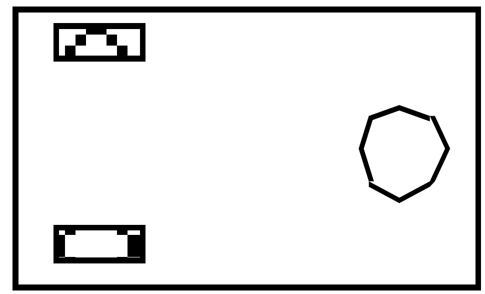
Mobile Robot Kinematics – e.g. different arrangements of wheels

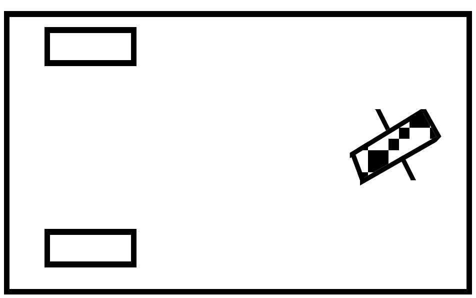
- Two wheels

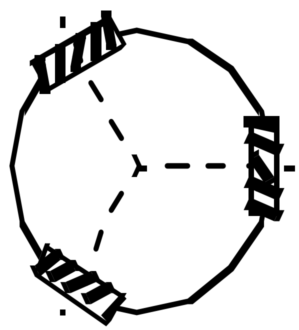


- Three wheels

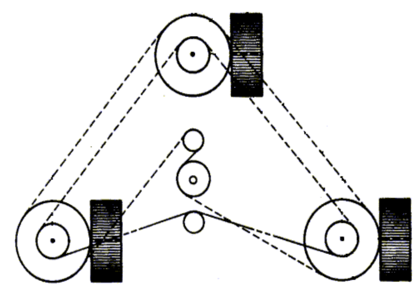








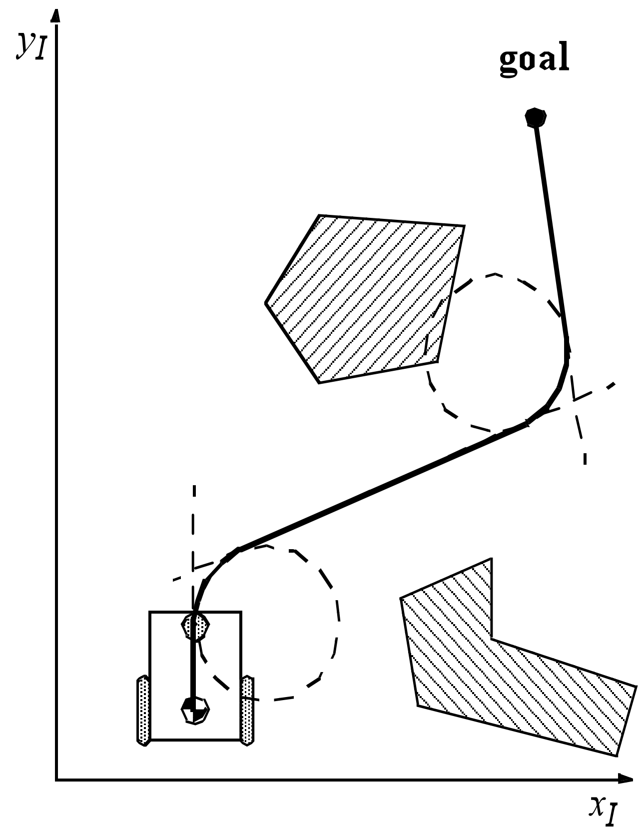
Omnidirectional Drive



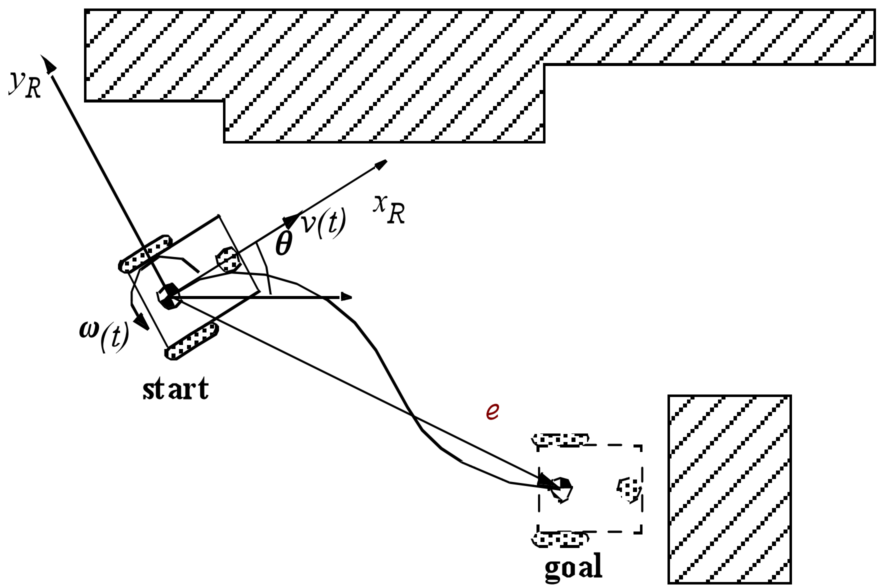
Synchro Drive

Motion Control: Open Loop Control

- trajectory (path) divided in motion segments of clearly defined shape:
  - straight lines and segments of a circle.
- control problem:
  - pre-compute a smooth trajectory based on line and circle segments



Motion Control: Feedback Control, Problem Statement



- Find a control matrix K, if exists

$$K = \begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \end{bmatrix}$$

- with $k_{ij}=k(t,e)$
- such that the control of $v(t)$ and $\omega(t)$

$$\begin{bmatrix} v(t) \\ \omega(t) \end{bmatrix} = K \cdot e = K \cdot {}^{R}\begin{bmatrix} x \\ y \\ \theta \end{bmatrix}$$

- drives the error e to zero.

$$\lim_{t \to \infty} e(t) = 0$$

Dealing with Uncertainty
Probabilistic Robotics

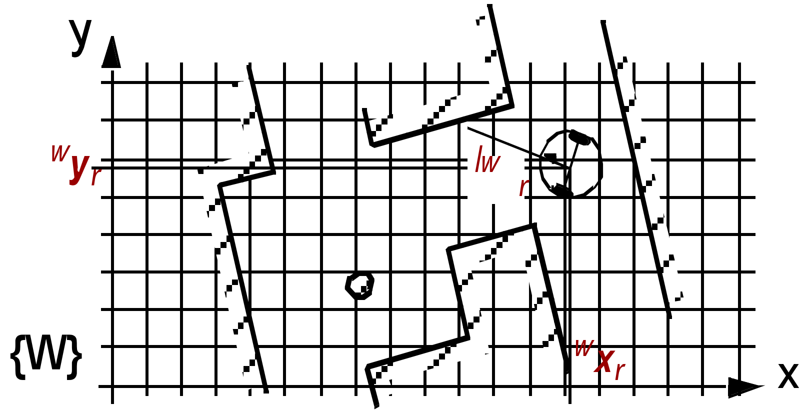
- Taking into account uncertainty of sensors and actions
- Localization in the presence of uncertainty,
- Map building

Robot Perception

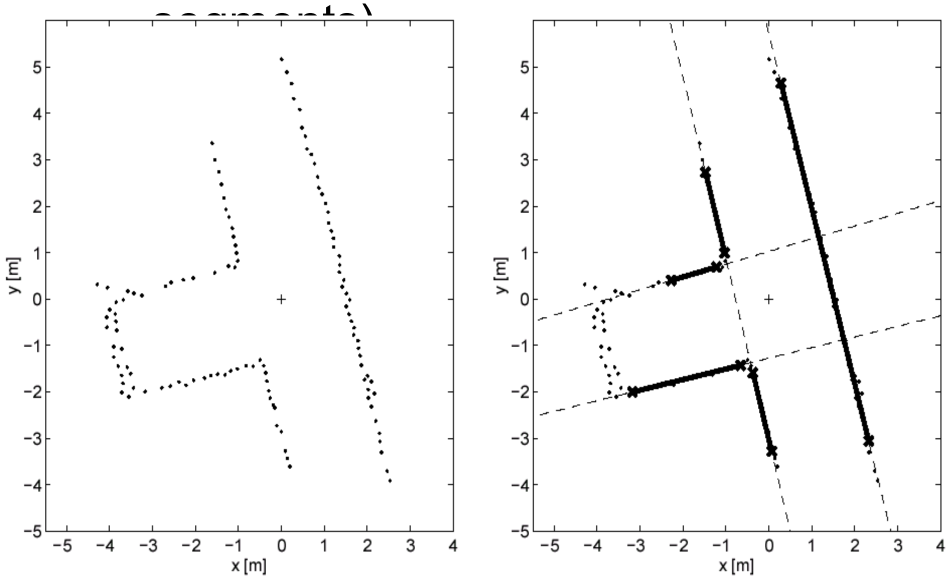
- How to process information from sensors
- Visual Sensing
- Range Sensing

Methods for Localization: The Quantitative Metric Approach

1. A priori Map: Graph, metric

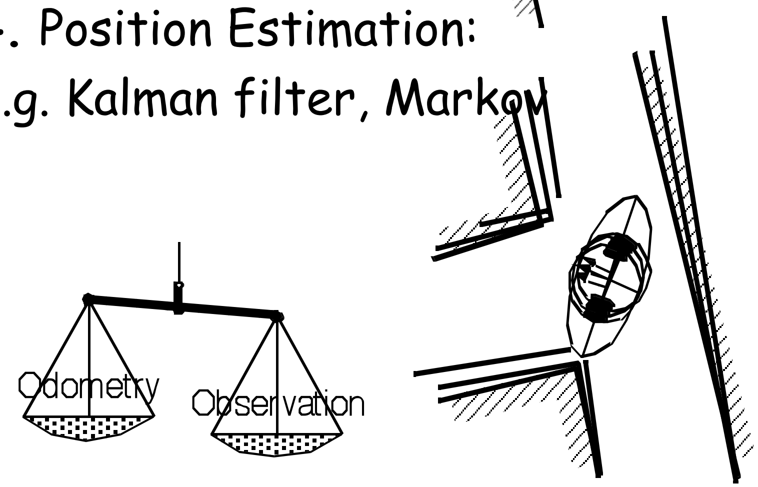


2. Feature Extraction (e.g. line segments)



3. Matching:
Find correspondence of features

4. Position Estimation:
e.g. Kalman filter, Markov



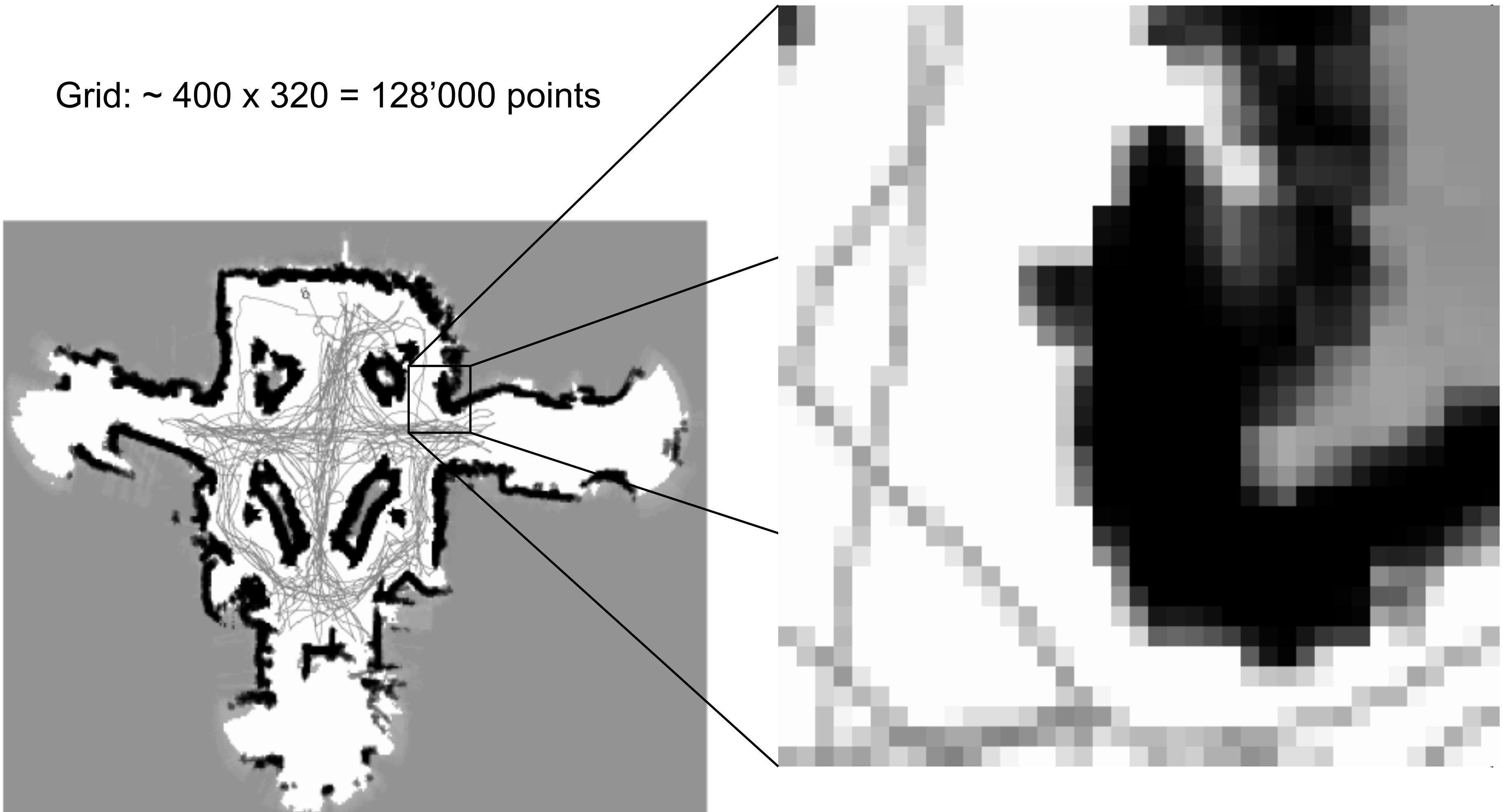
- representation of uncertainties
- optimal weighting acc. to a priori statistics

Courtesy K. Arras

Grid-Based Metric Approach

- Grid Map of the Smithsonian's National Museum of American History in Washington DC. (Courtesy of Wolfram Burger et al.)

Grid: ~ 400 x 320 = 128'000 points

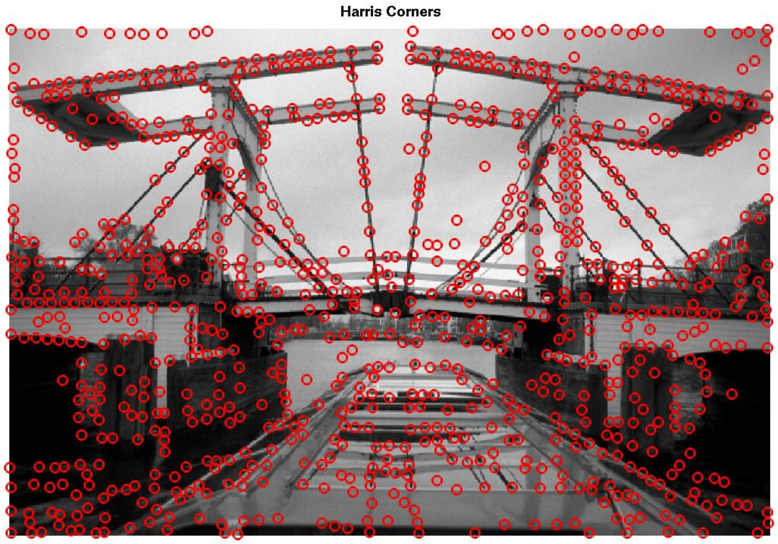


Courtesy S. Thrun, W. Burgard

Robot Perception (Image features)

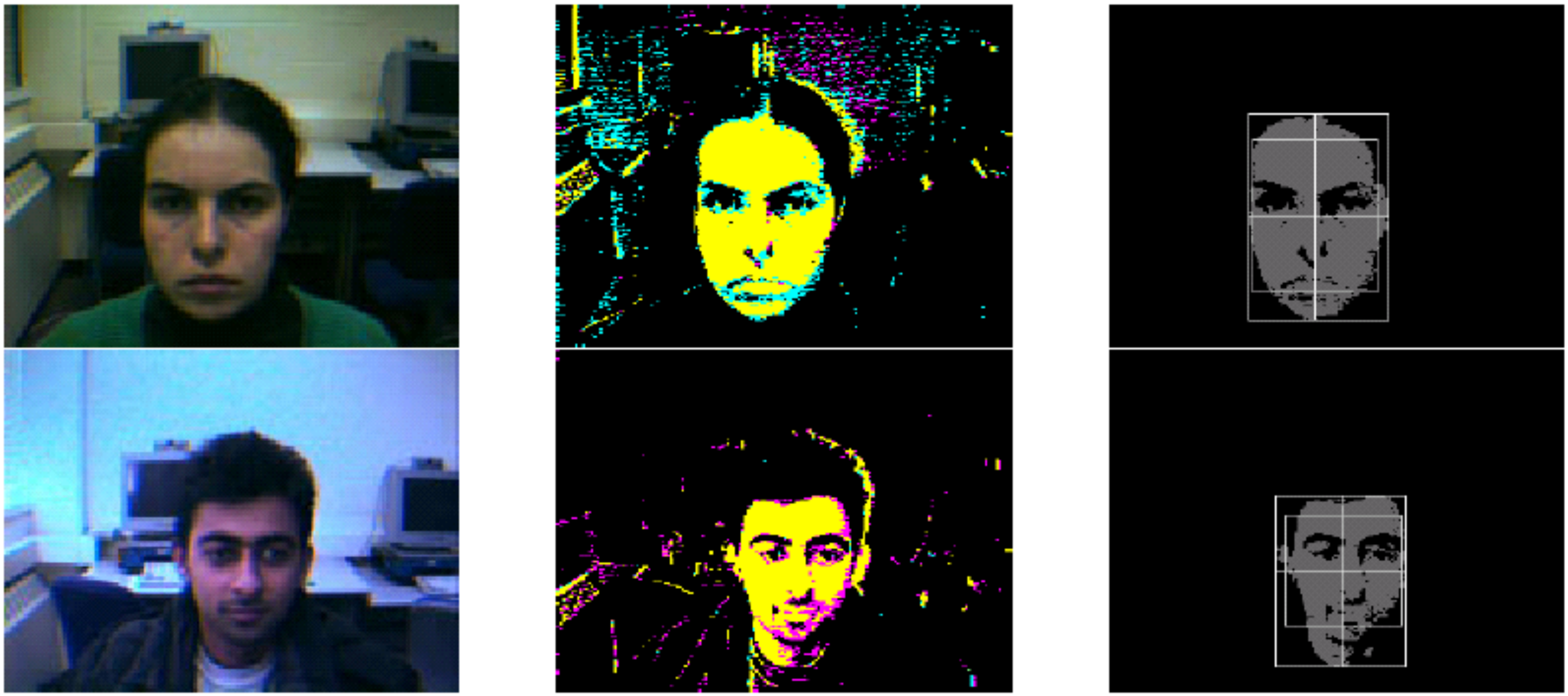
Original image

Strong + connected weak edges



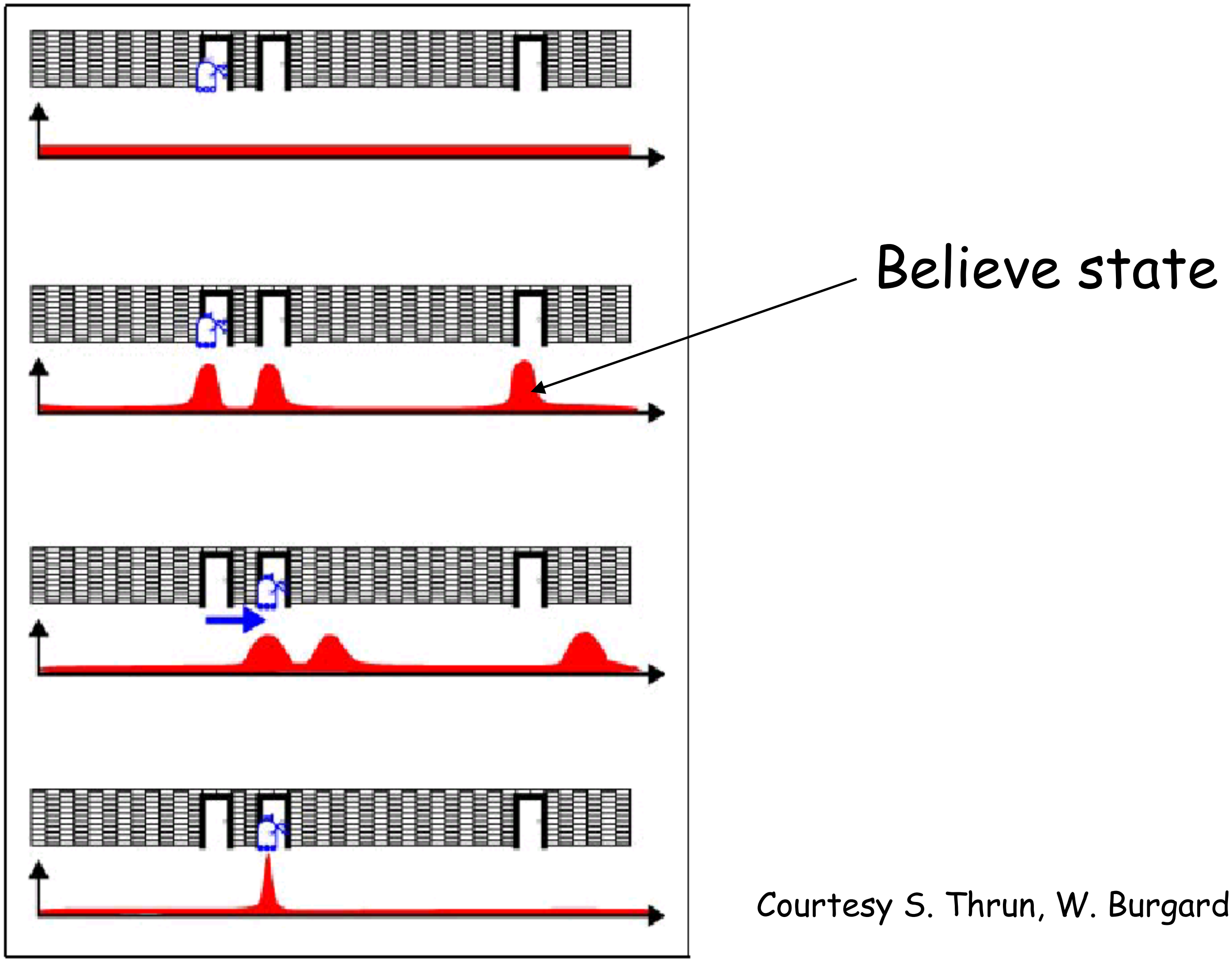
Interest points

courtesy of G. Loy



Jana Kosecka

Gaining Information through motion: (Multi-hypotheses tracking)



Courtesy S. Thrun, W. Burgard

Markov Localization (4): Applying probability theory to robot localization

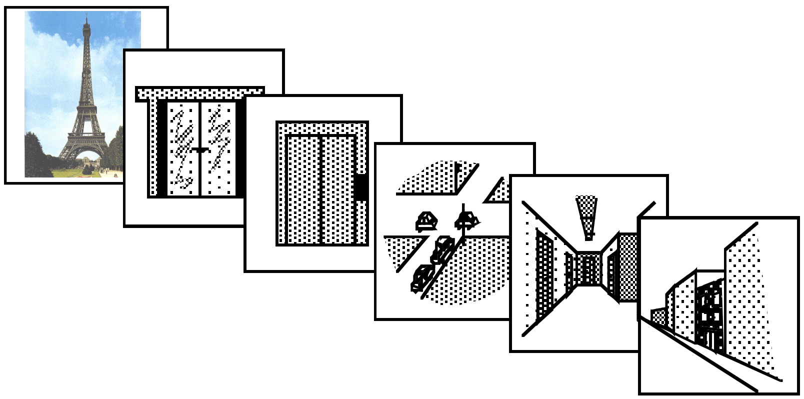
- Bayes rule: $$p(A|B) = \frac{p(B|A)p(A)}{p(B)}$$
  - Map from a belief state and a action to new belief state (ACT):

$$p(l_t|o_t) = \int p(l_t|l'_{t-1}, o_t)p(l'_{t-1})dl'_{t-1}$$

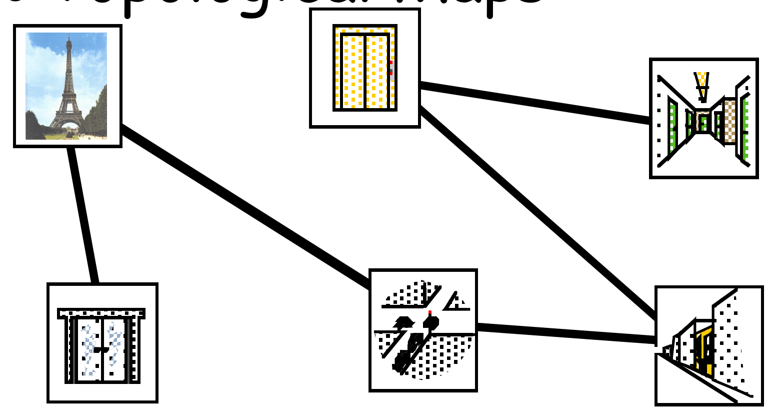
  - Summing over all possible ways in which the robot may have reached l.
- Markov assumption: Update only depends on previous state and its most recent actions and perception.

Environment Representation and Modeling

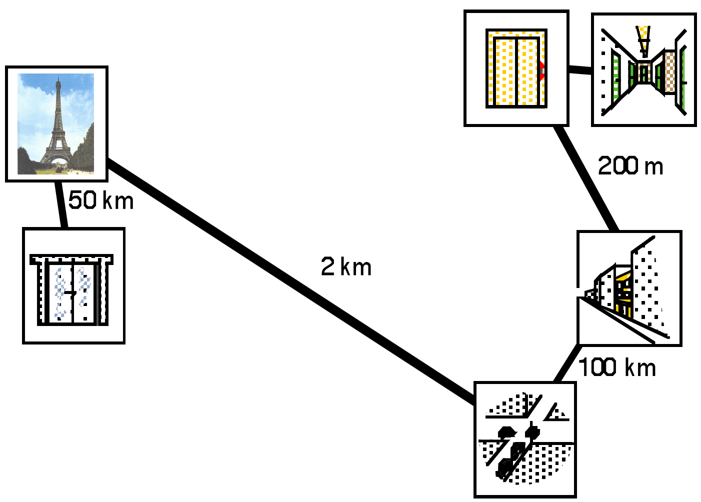
- Recognizable Locations



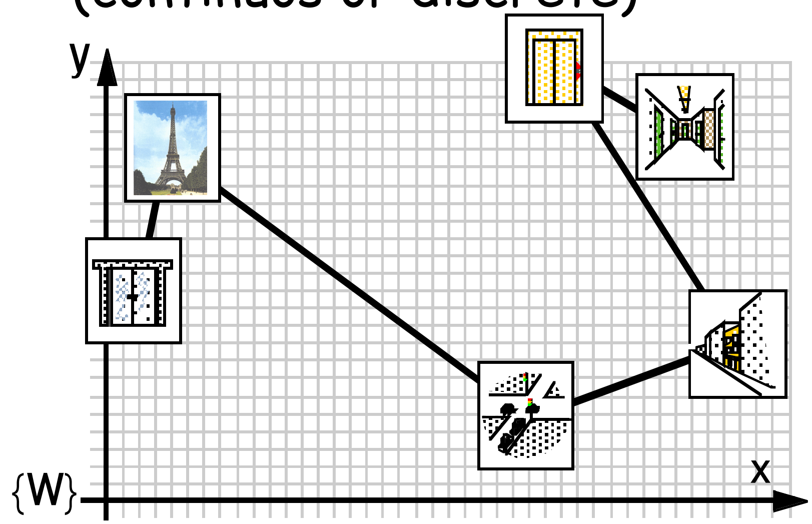
- Topological Maps



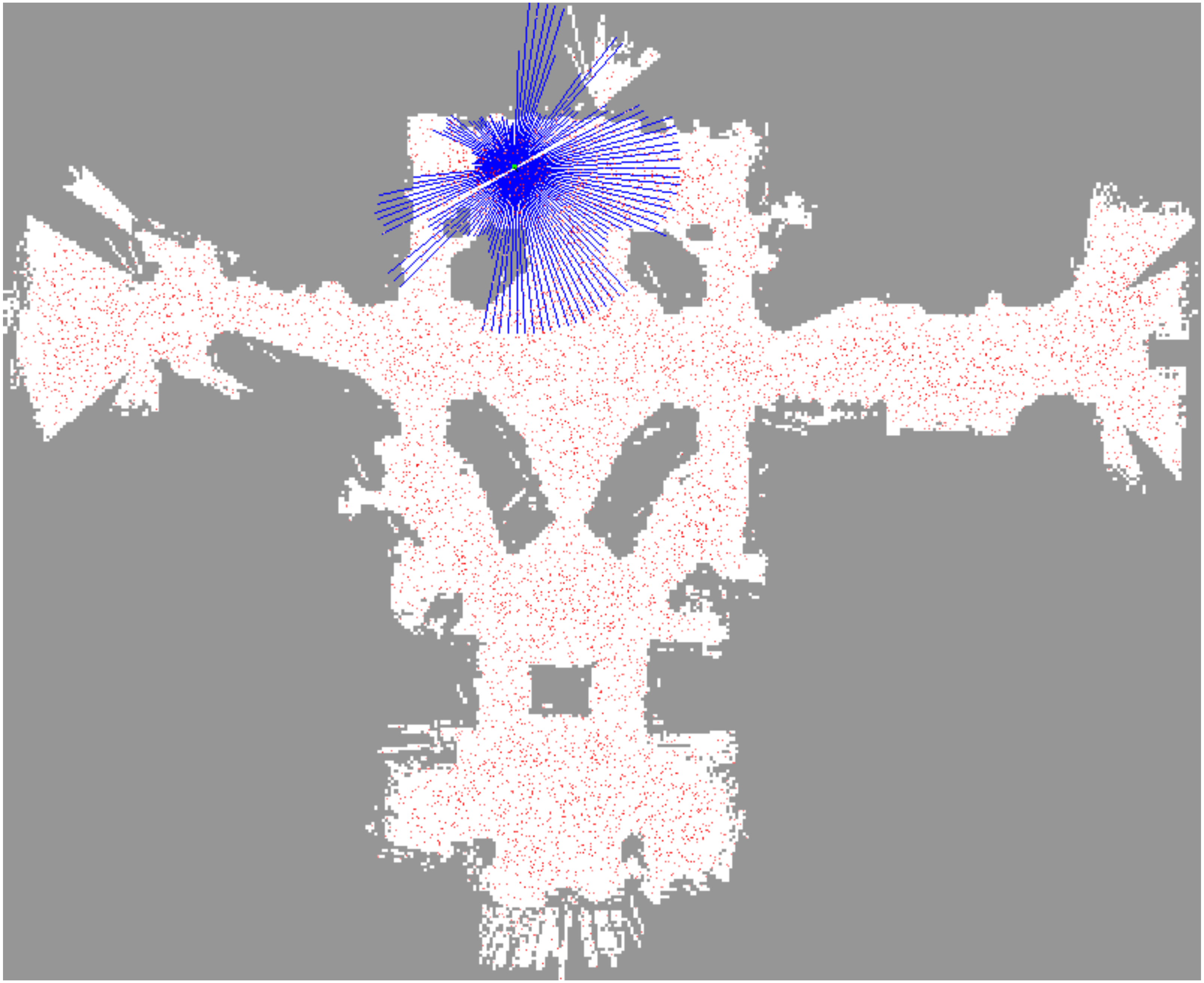
- Metric Topological Maps

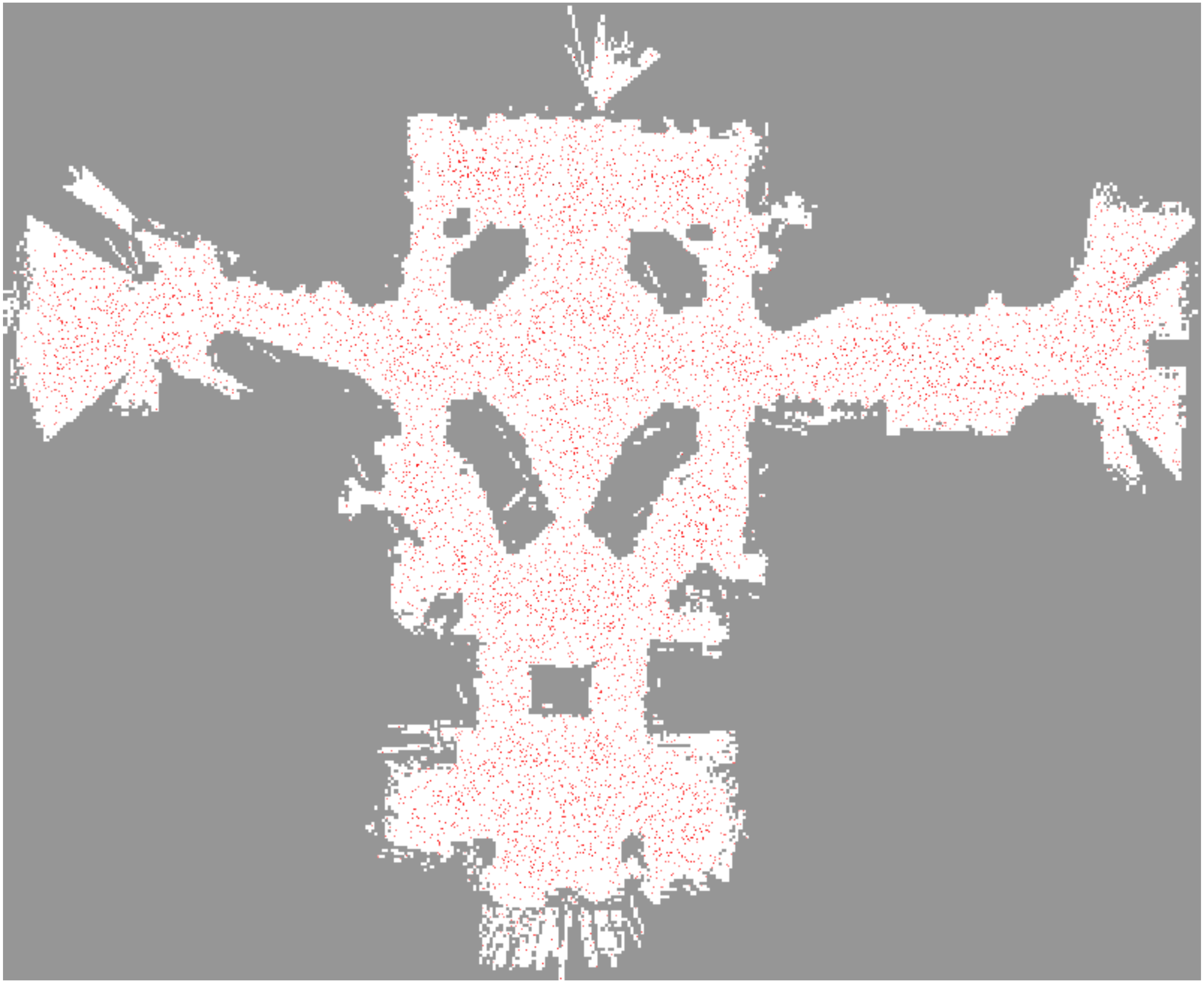


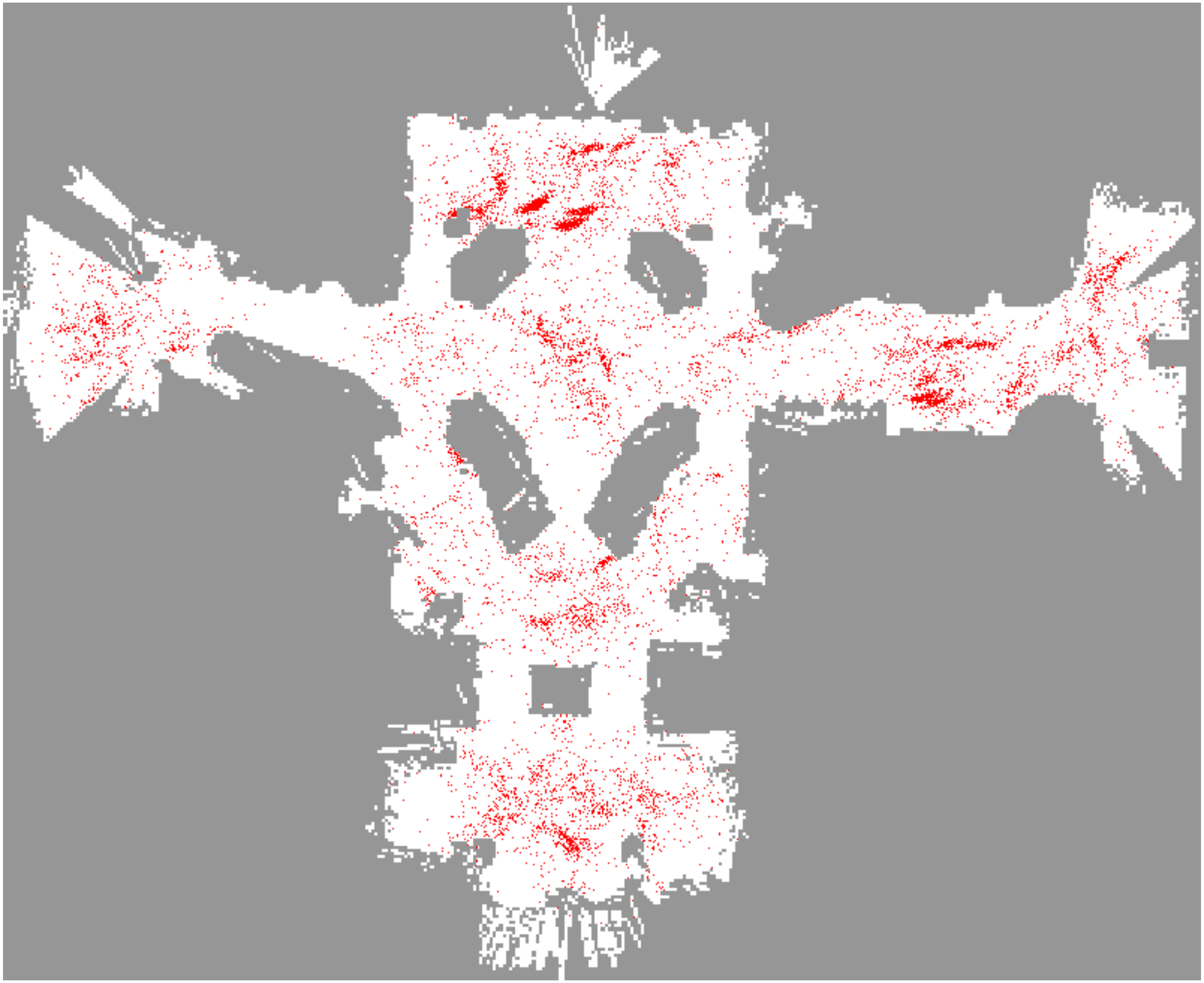
- Fully Metric Maps
  (continuos or discrete)

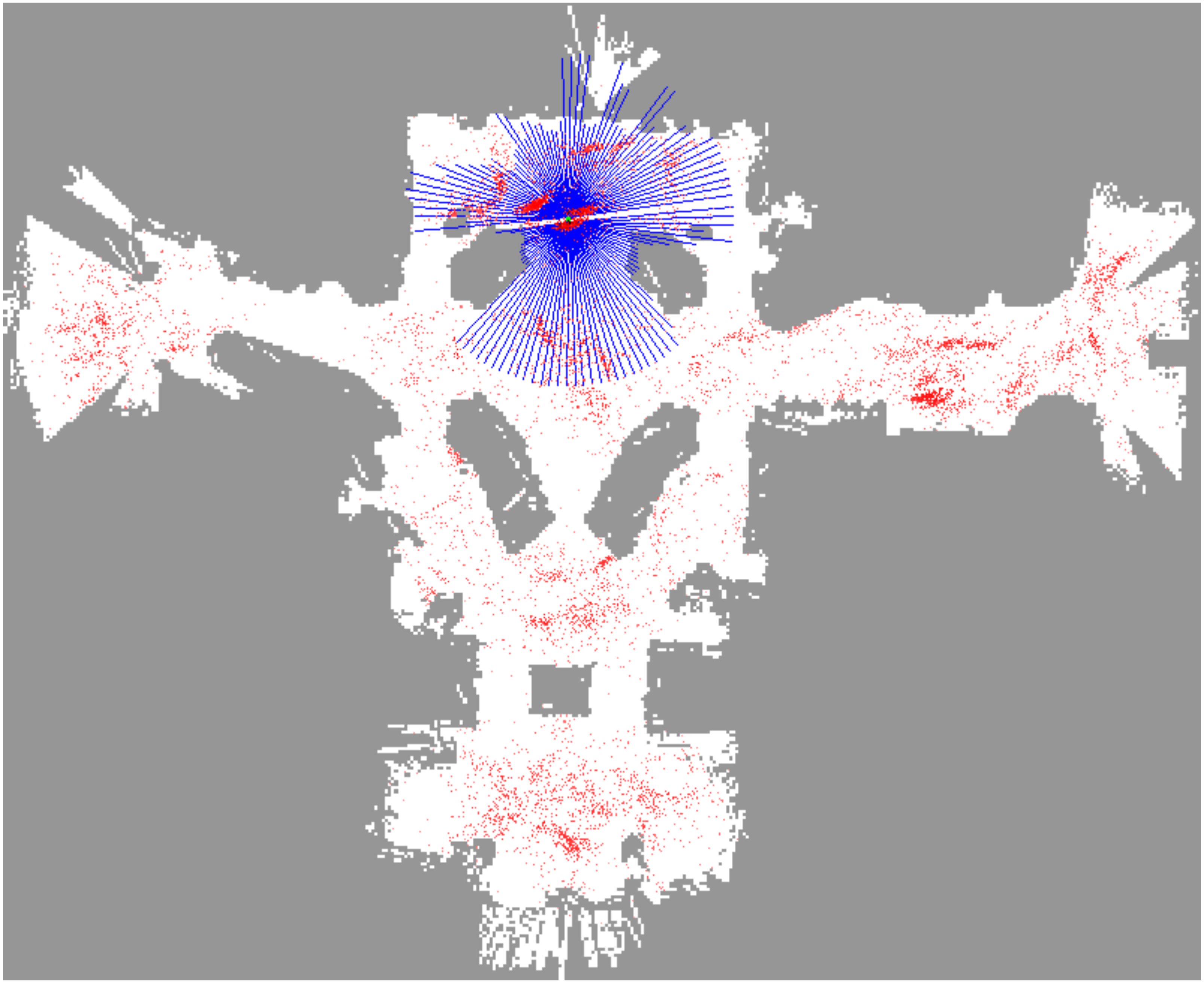


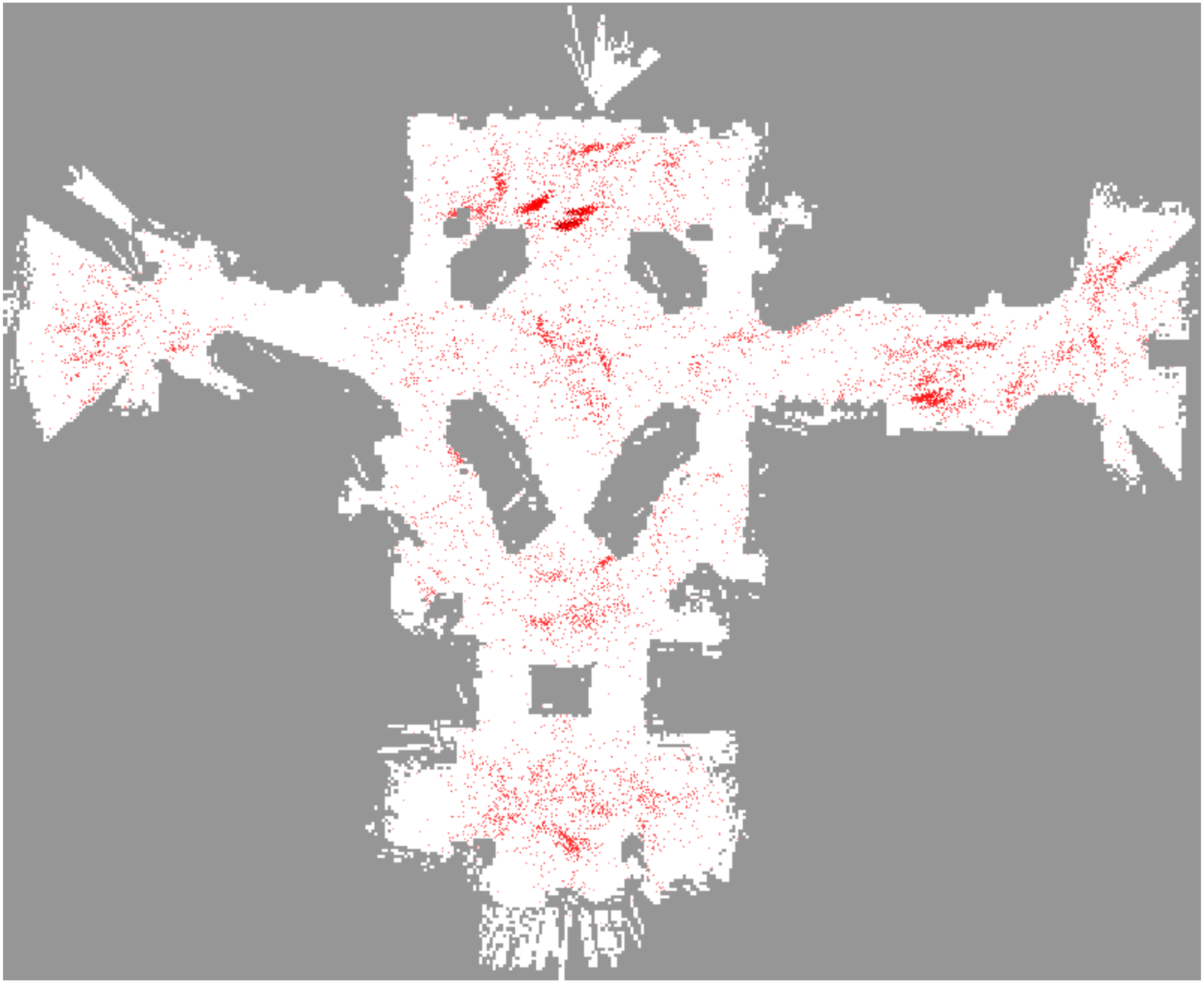
Courtesy K. Arras

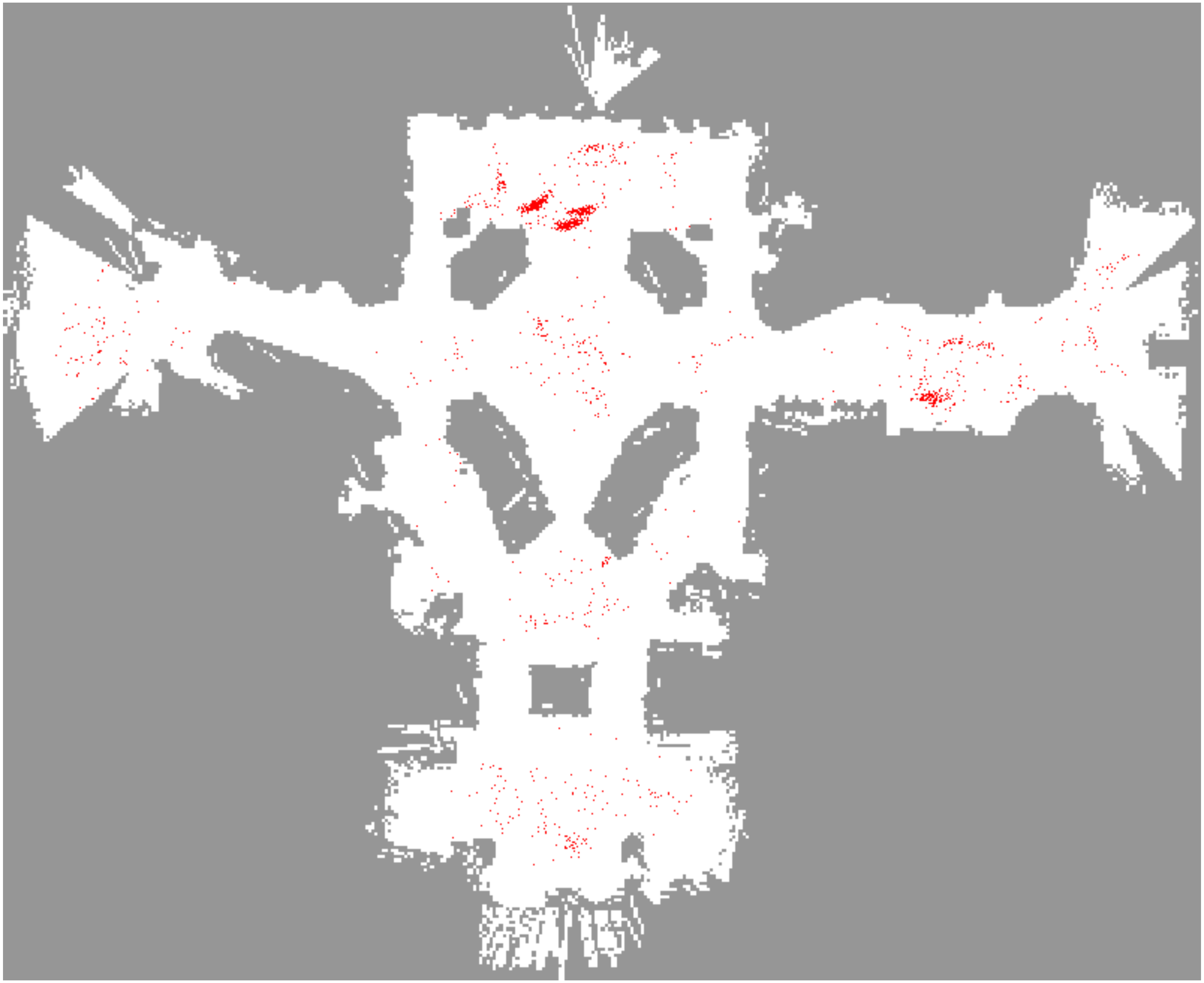


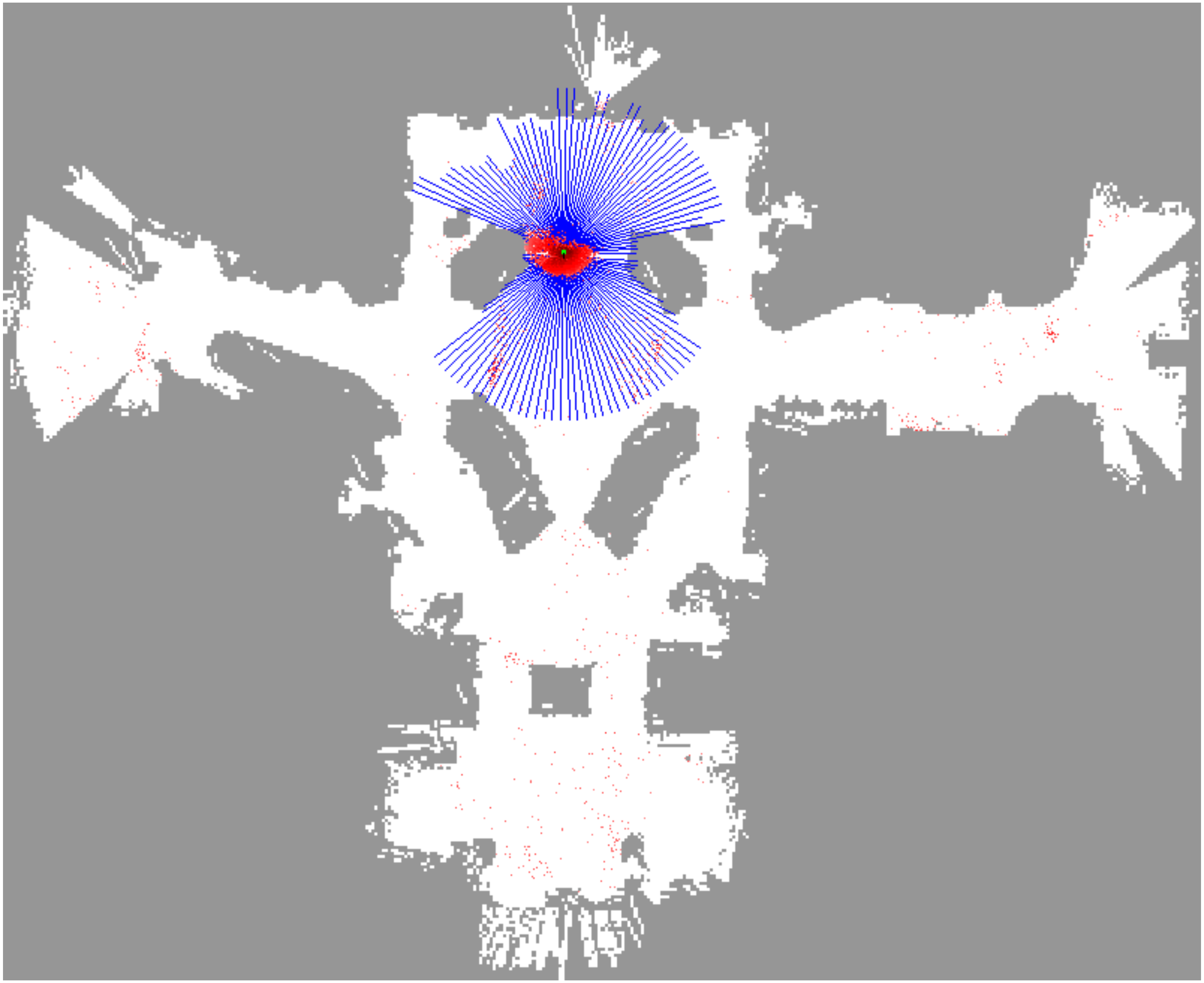


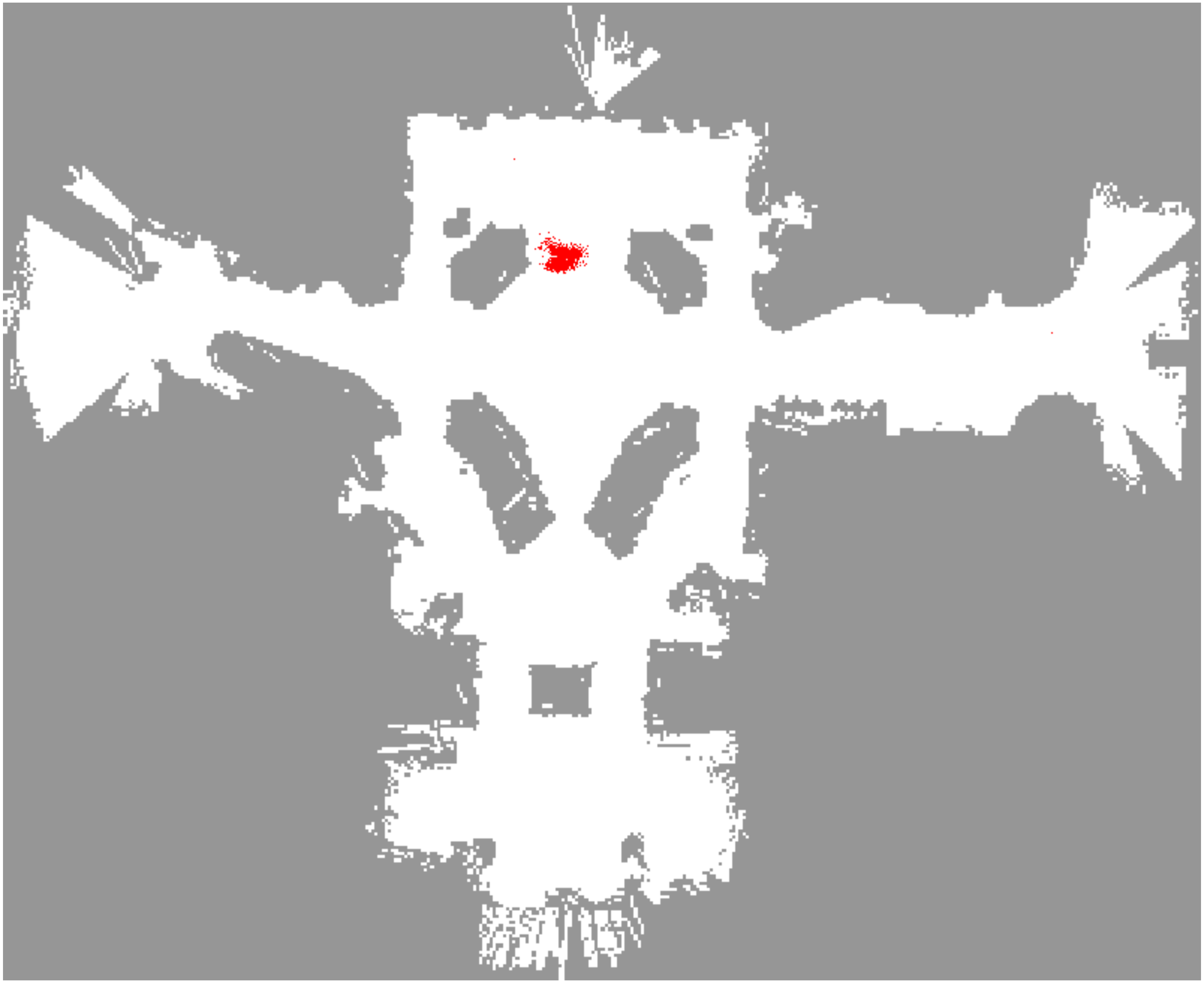


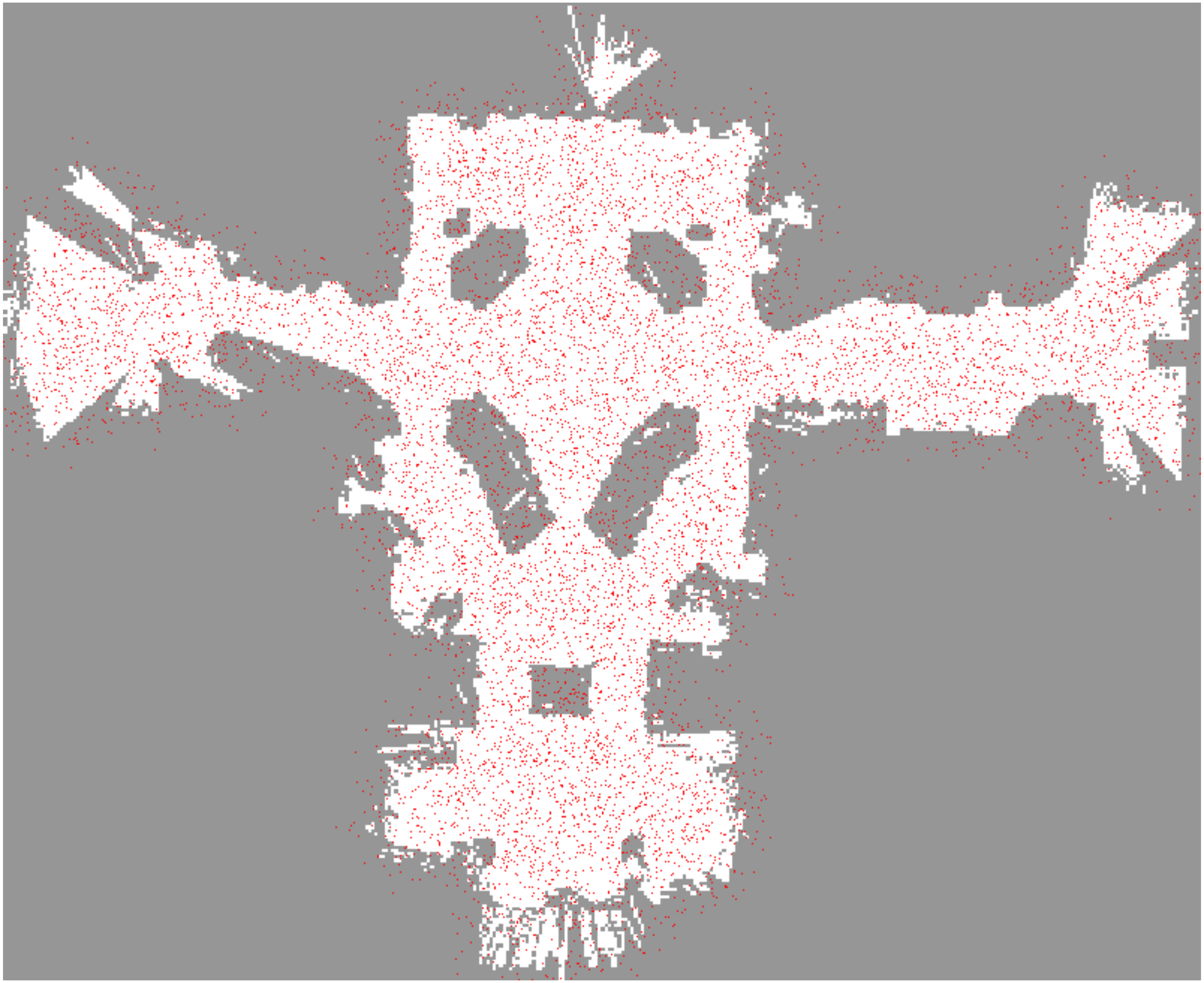


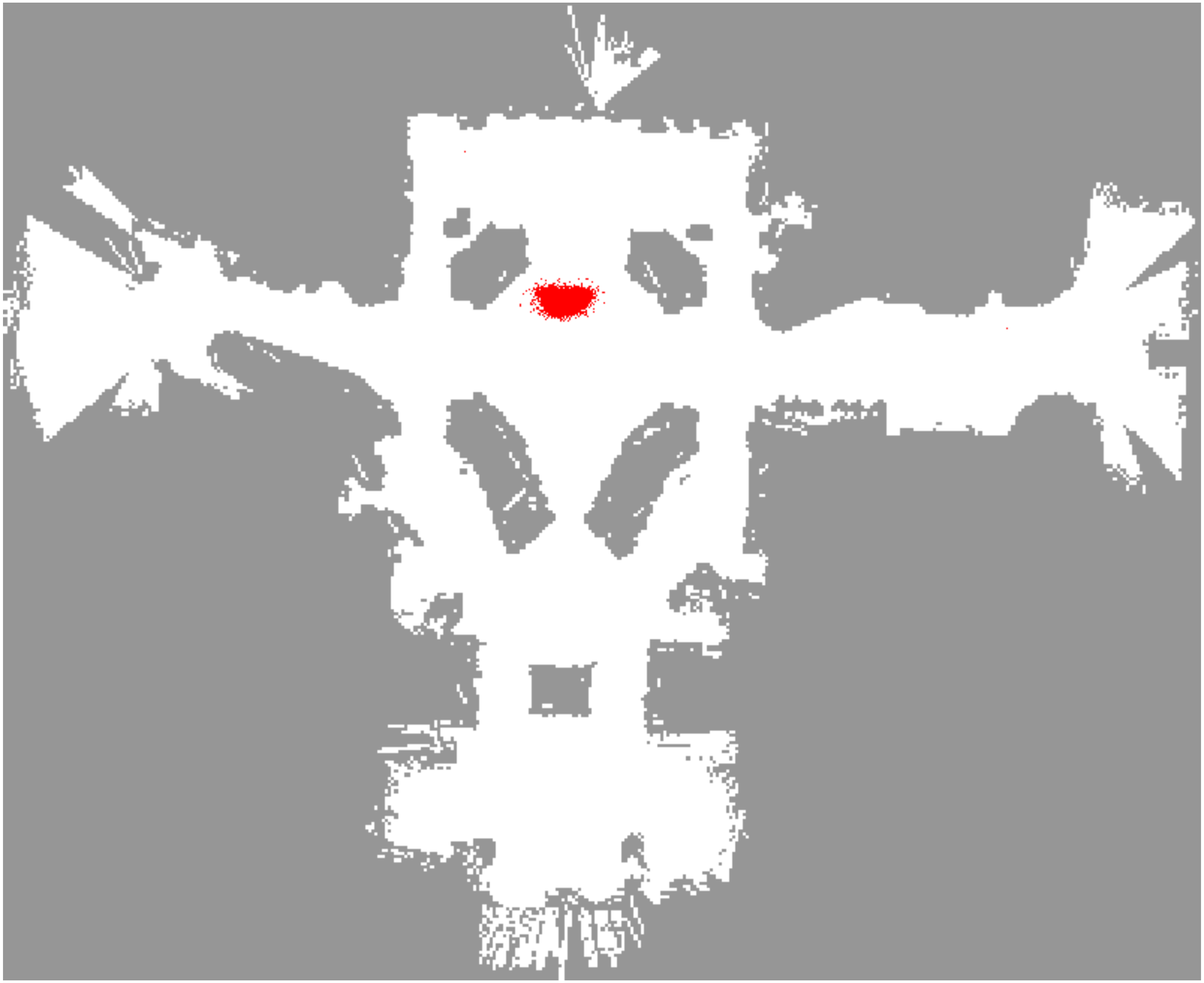


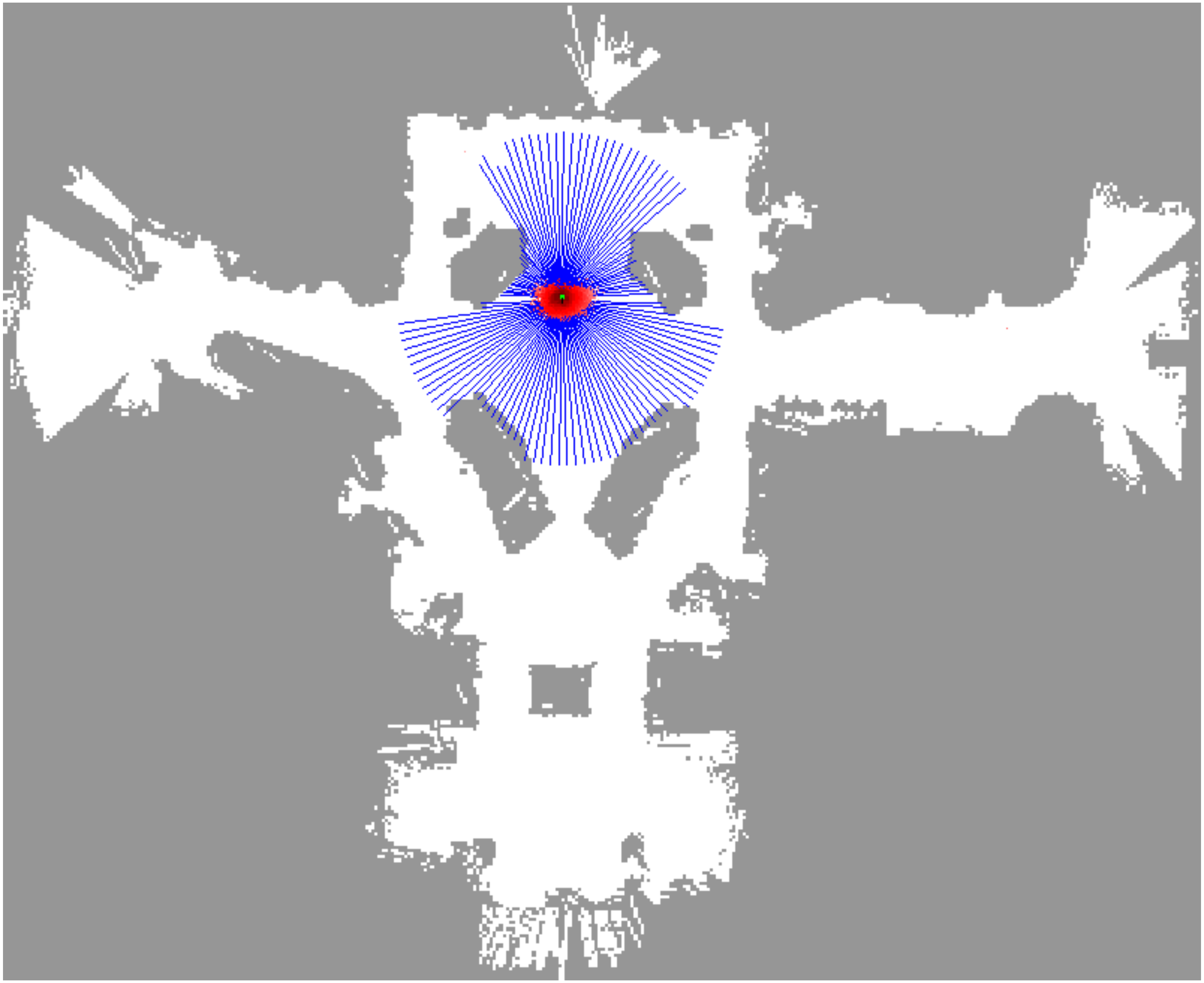


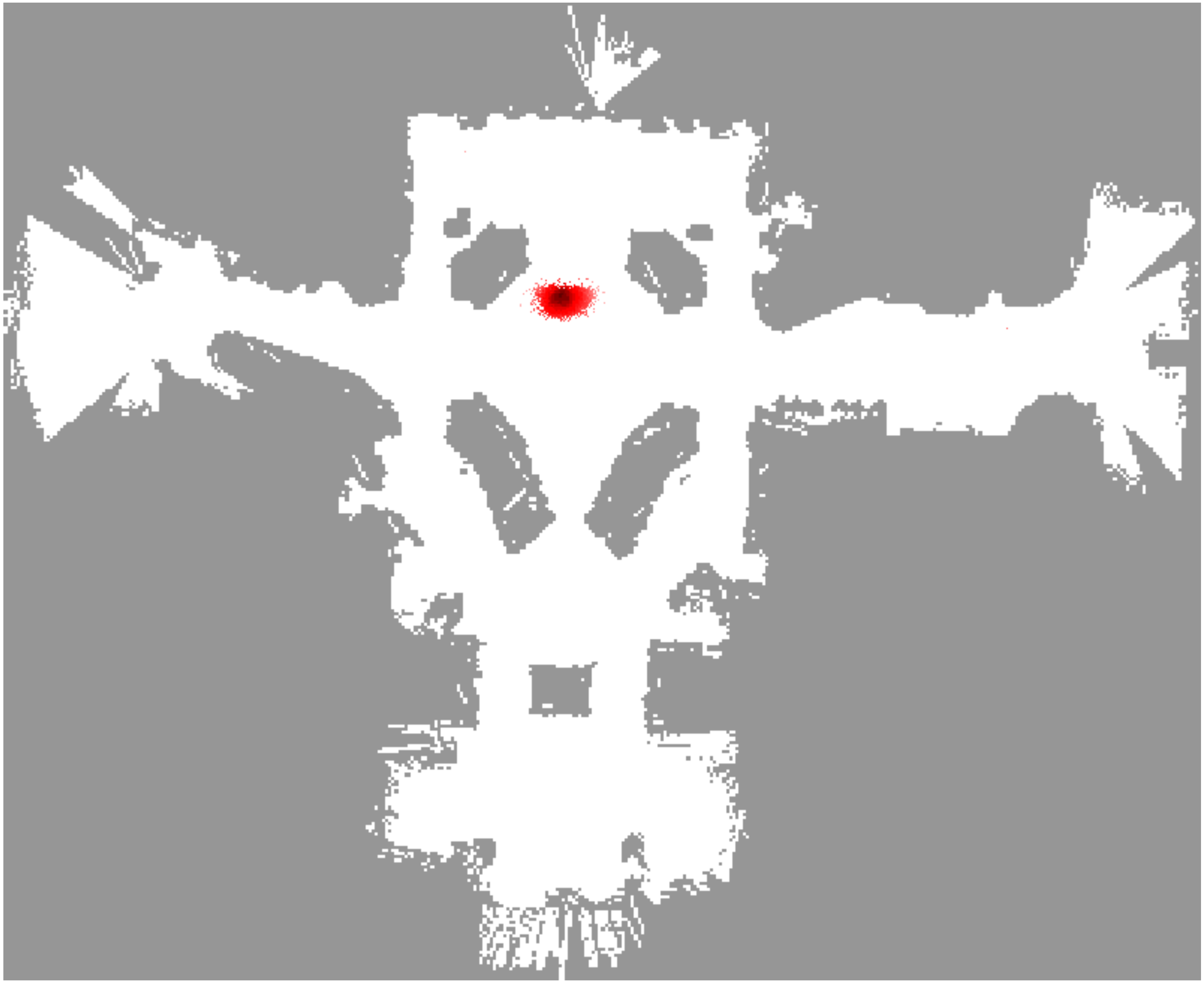


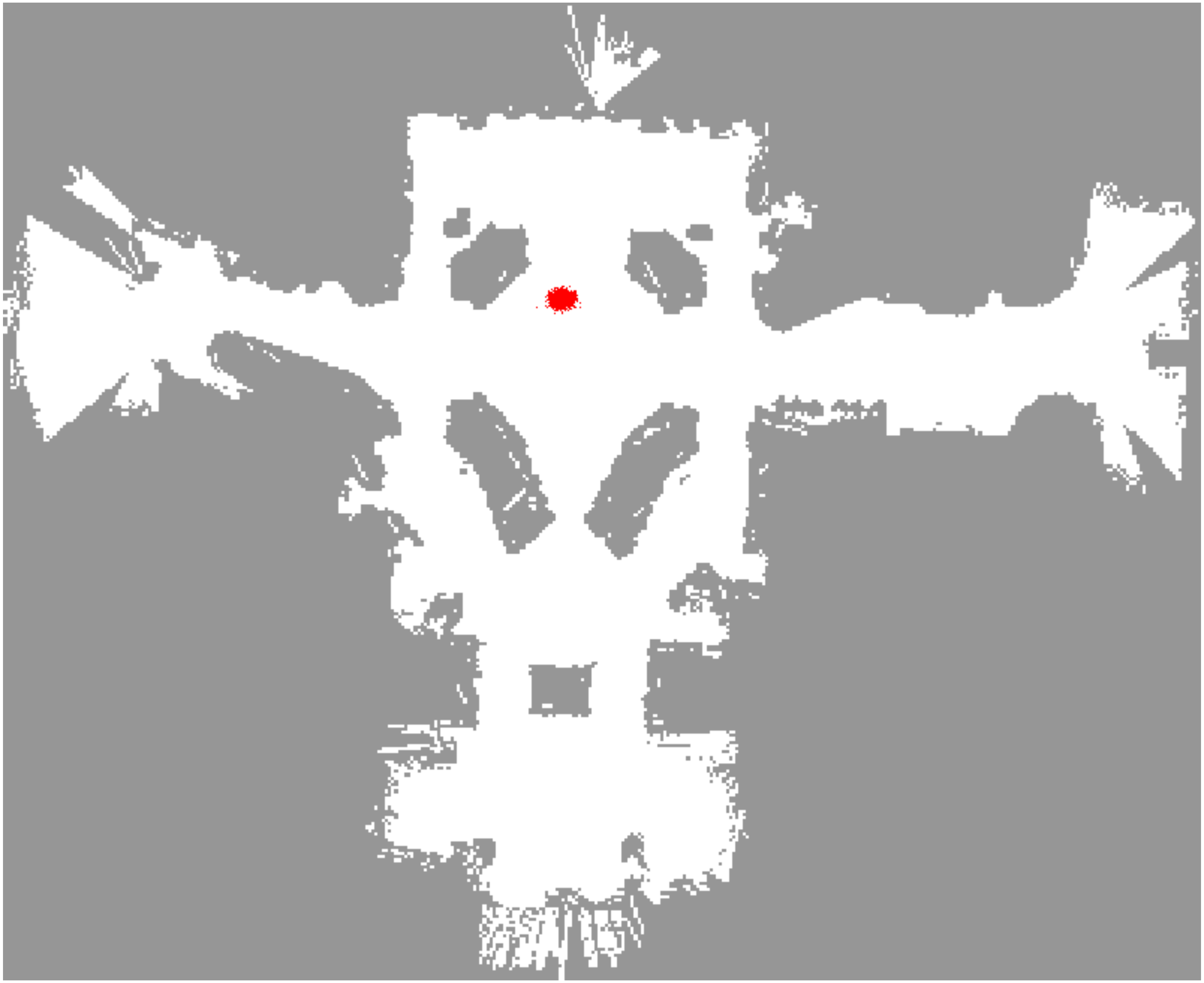






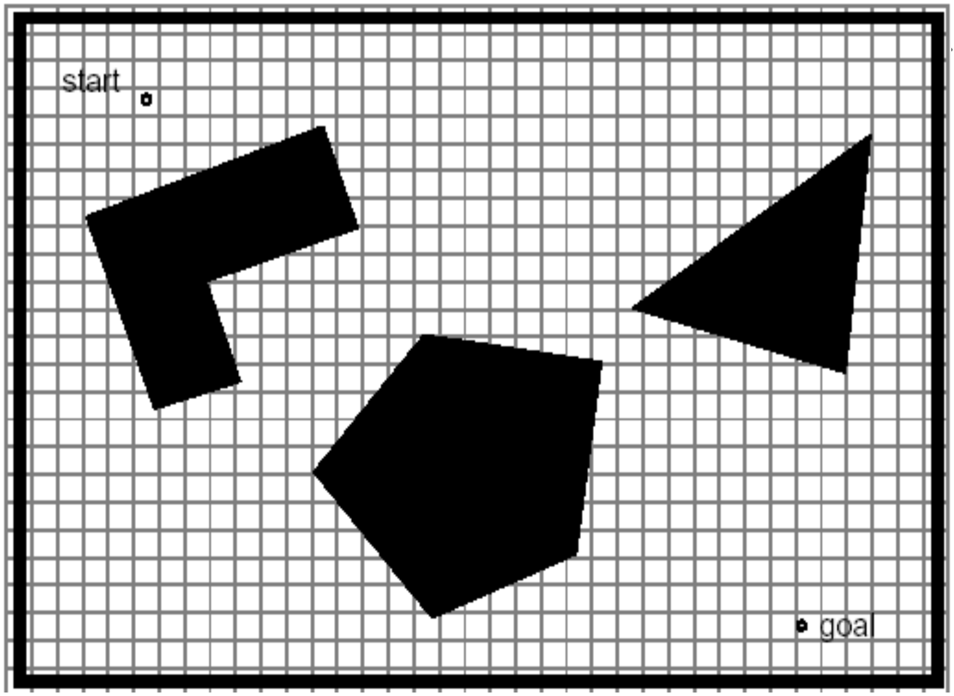


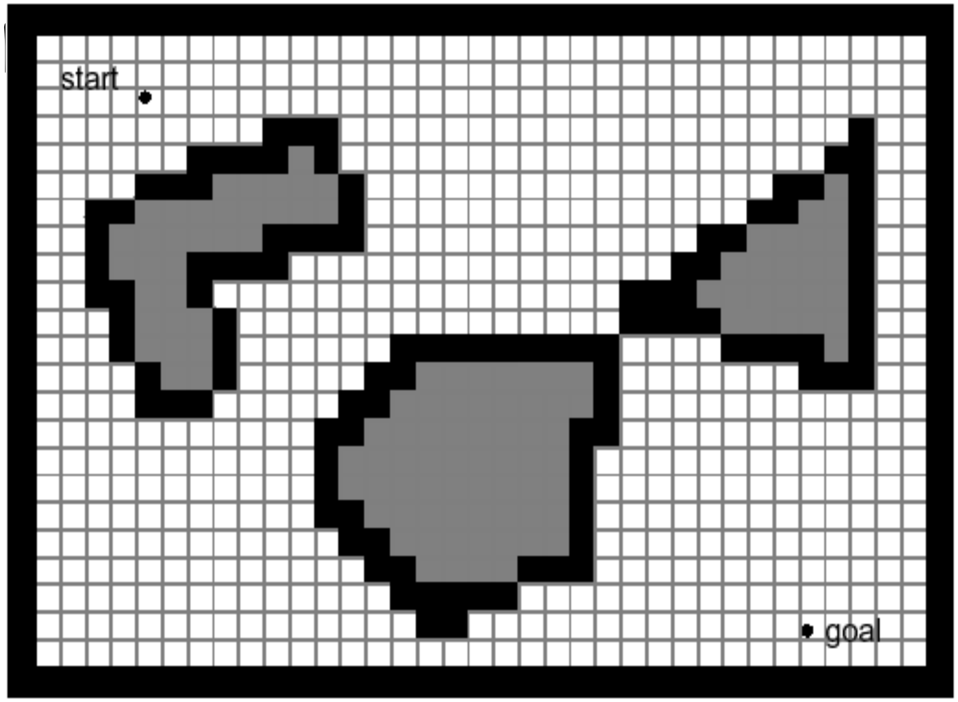




Map Representation: Decomposition (2)

- Fixed cell decomposition





Motion Planning

- Algorithms for determining movements of the robot in cluttered environments
- General techniques – 1st assumption – the environment is known
- Continuous representations of environments
- Discrete representations of the environments
- Deterministic methods – optimality, feasibility guarantees

- Motion planning for mobile robots, arbitrary shaped parts, articulated structures

- Randomized algorithms for motion planning

Reinforcement Learning

- How to improve performance over time from our own/systems experience
- Goal directed learning from interaction
- How to map situations to action to maximize reward