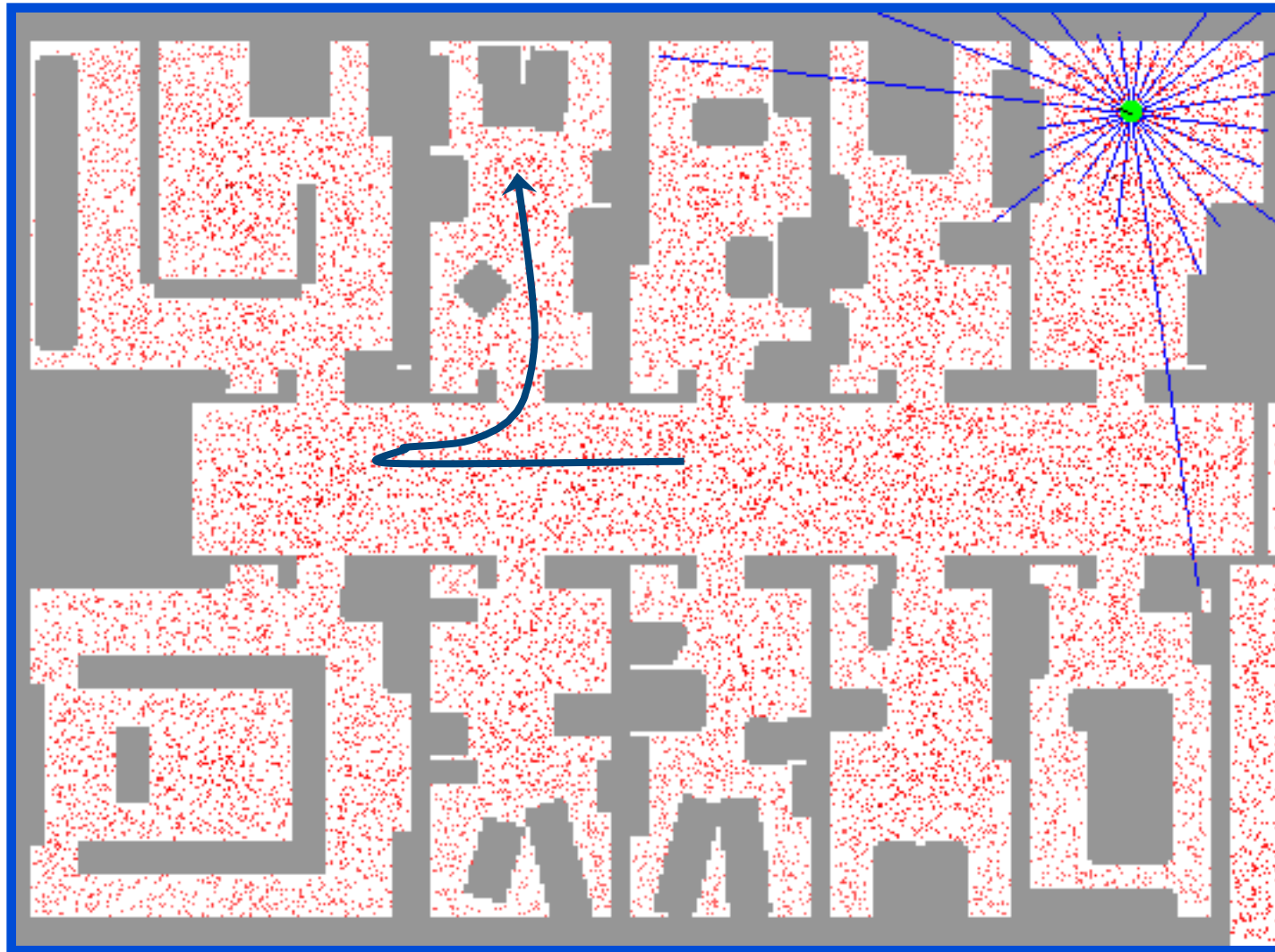


Probabilistic Robotics

Bayes Filter Implementations

Particle filters

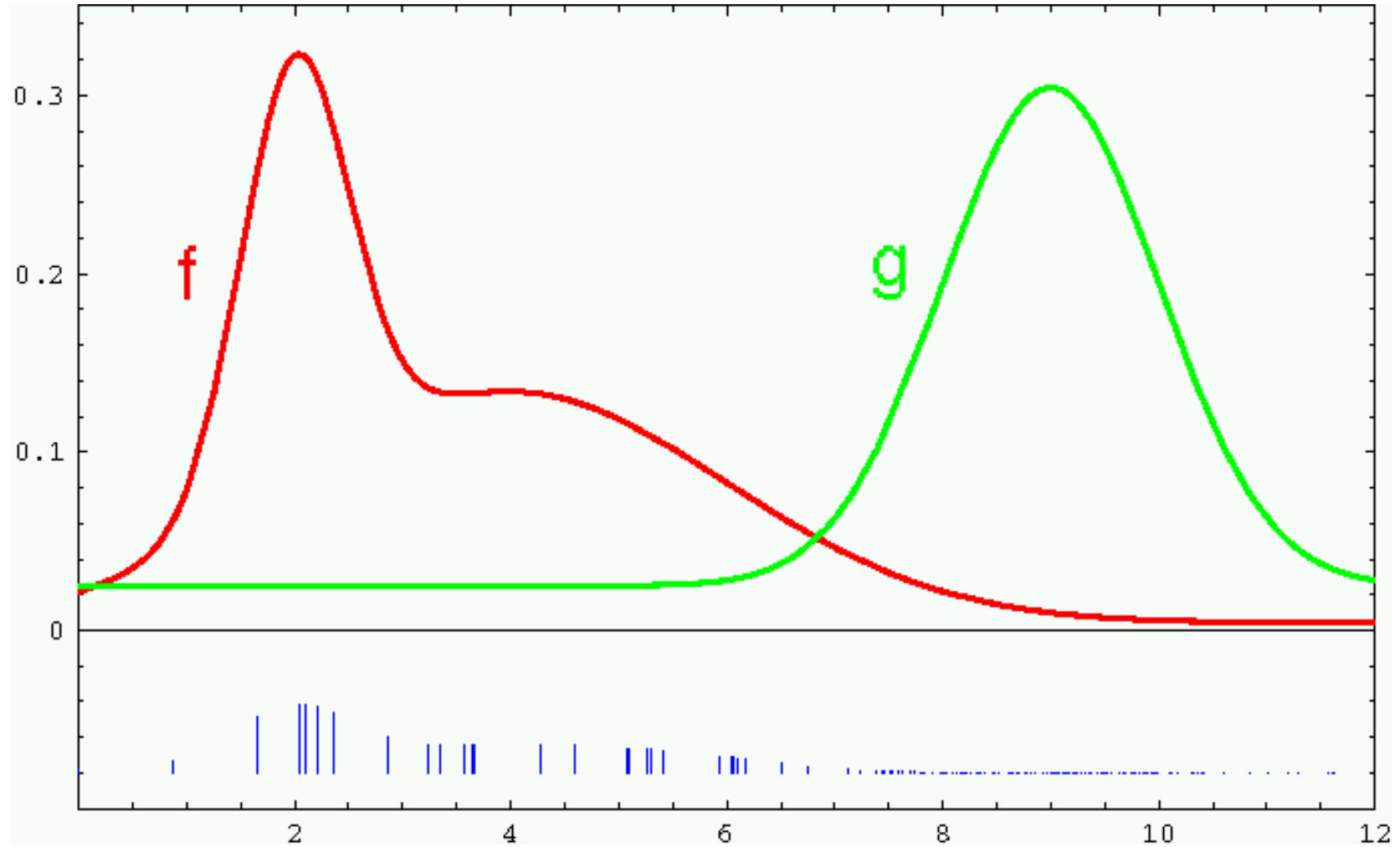
Sample-based Localization (sonar)



Particle Filters

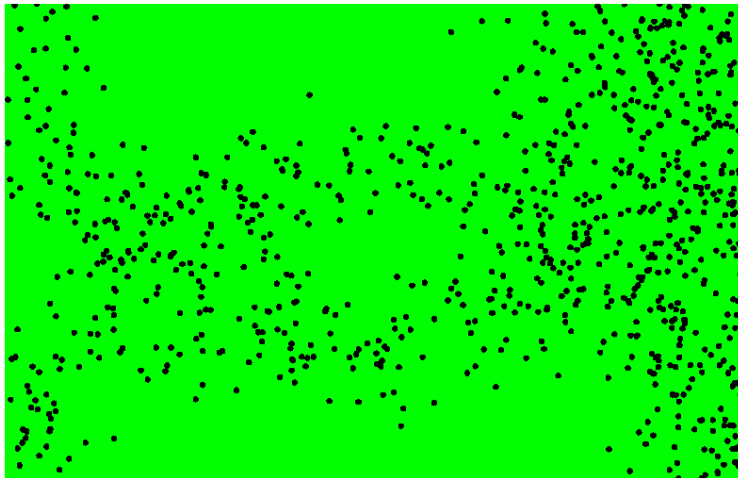
- Represent belief by random **samples**
- Estimation of **non-Gaussian, nonlinear** processes
- Monte Carlo filter, Survival of the fittest, Condensation, Bootstrap filter, Particle filter
- Filtering: [Rubin, 88], [Gordon et al., 93], [Kitagawa 96]
- Computer vision: [Isard and Blake 96, 98]
- Dynamic Bayesian Networks: [Kanazawa et al., 95]d

Importance Sampling



Weight samples: $w = f/g$

Importance Sampling with Resampling

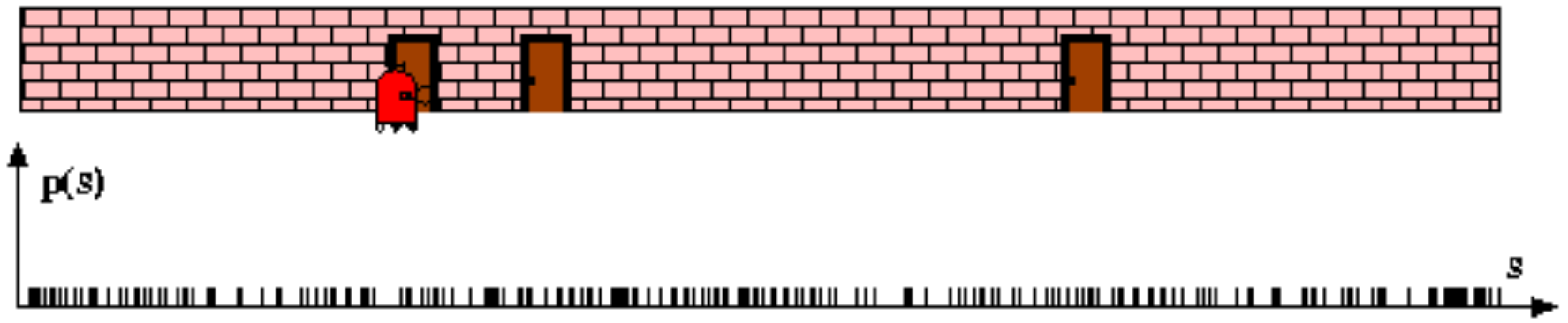


Weighted samples



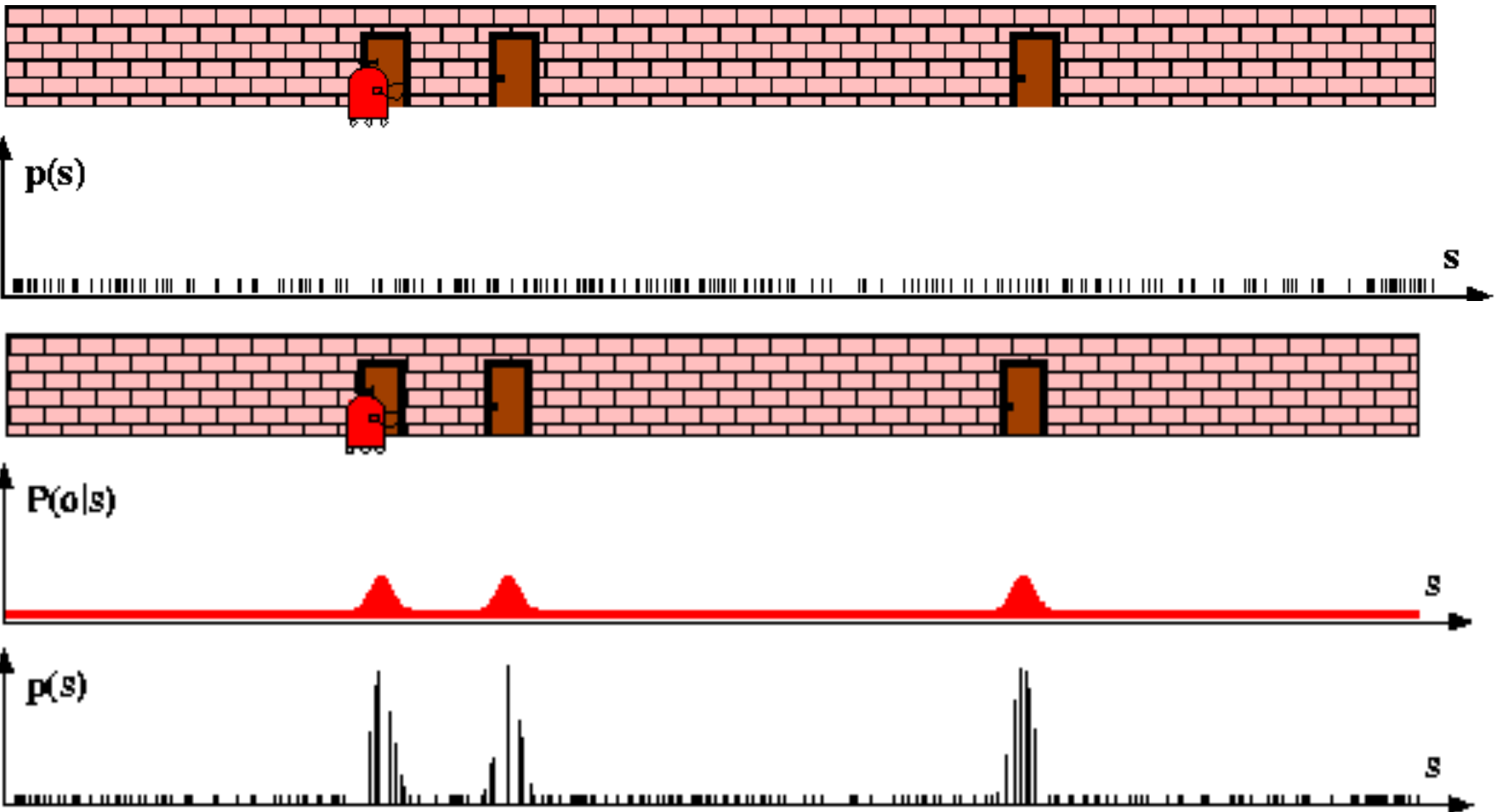
After resampling

Particle Filters



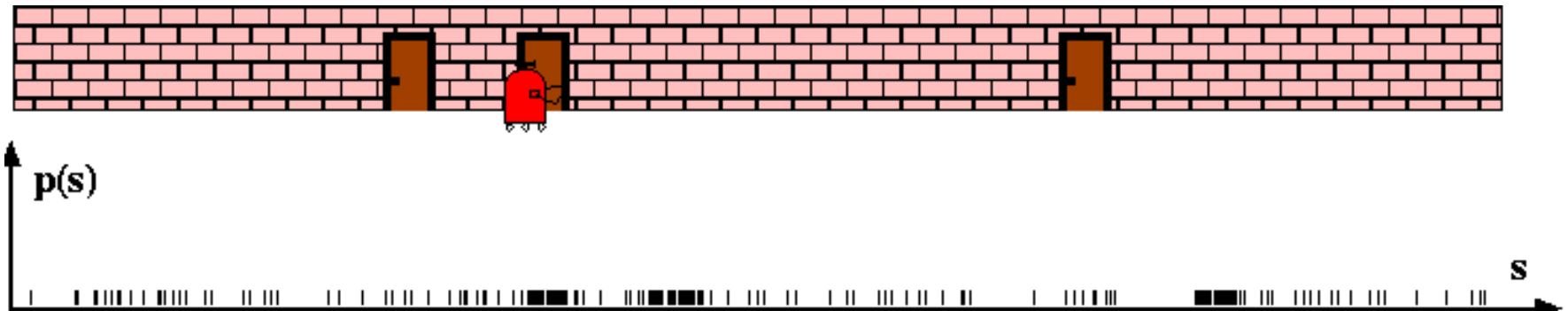
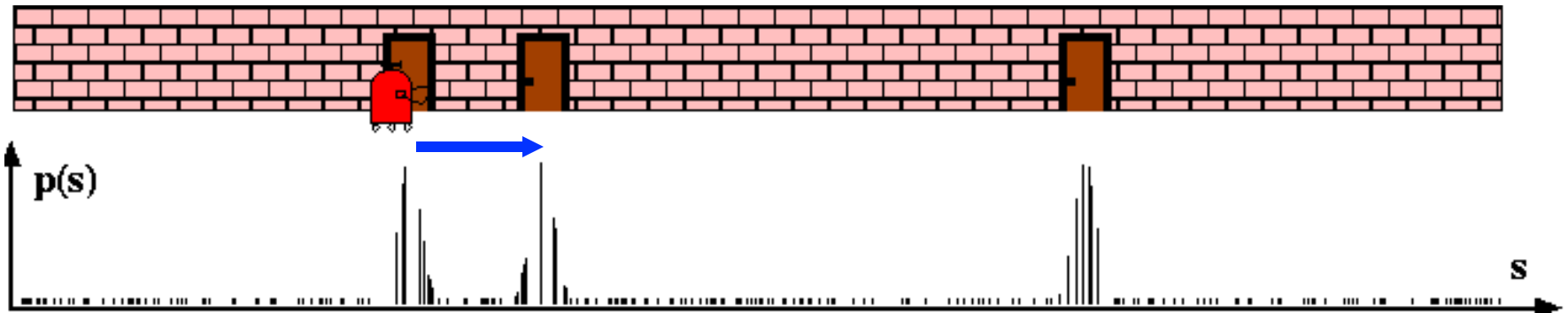
Sensor Information: Importance Sampling

$$\begin{aligned} Bel(x) &\leftarrow \alpha p(z|x) Bel^-(x) \\ w &\leftarrow \frac{\alpha p(z|x) Bel^-(x)}{Bel^-(x)} = \alpha p(z|x) \end{aligned}$$



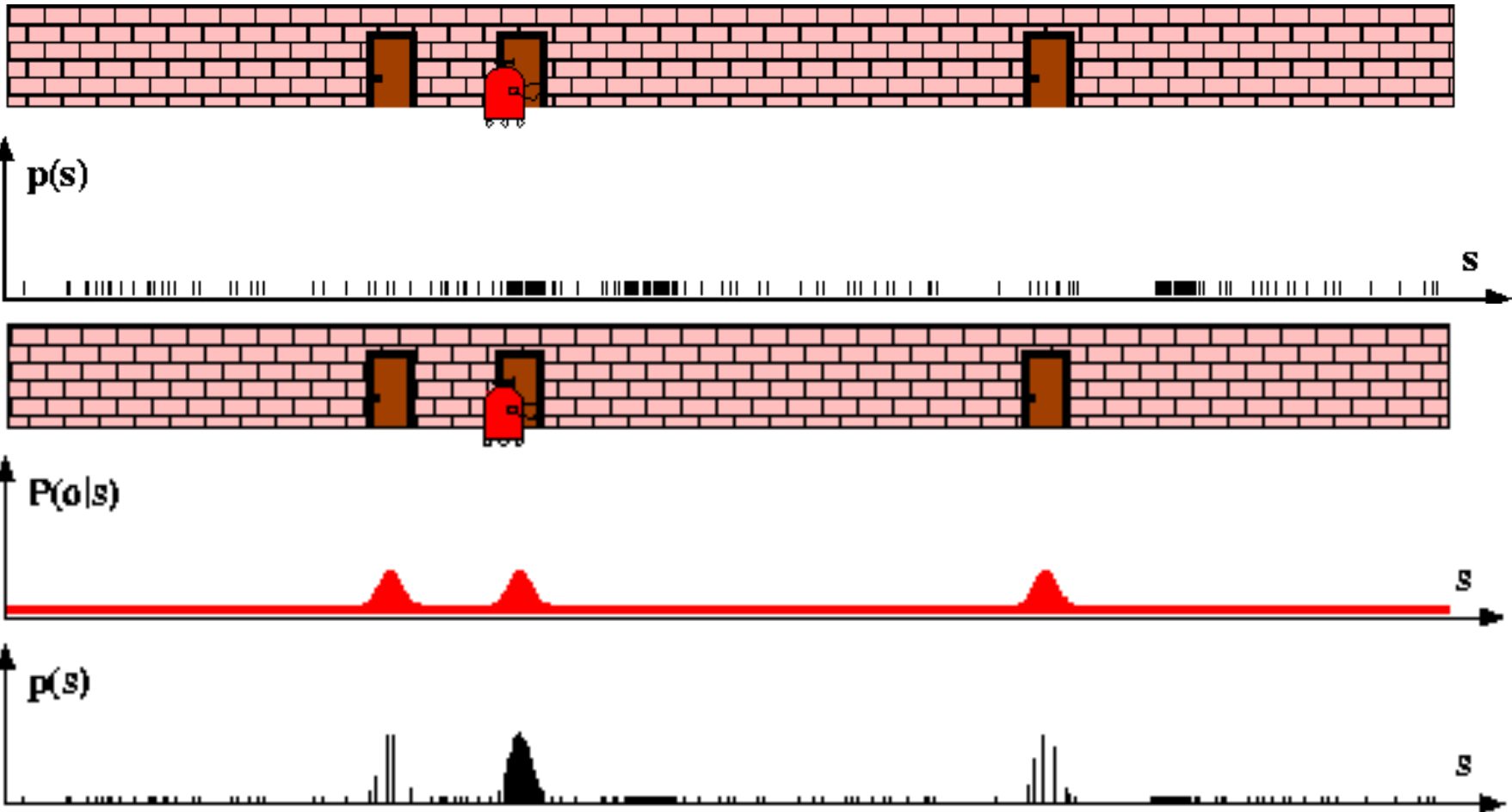
Robot Motion

$$Bel^-(x) \leftarrow \int p(x|u, x') Bel(x') dx'$$



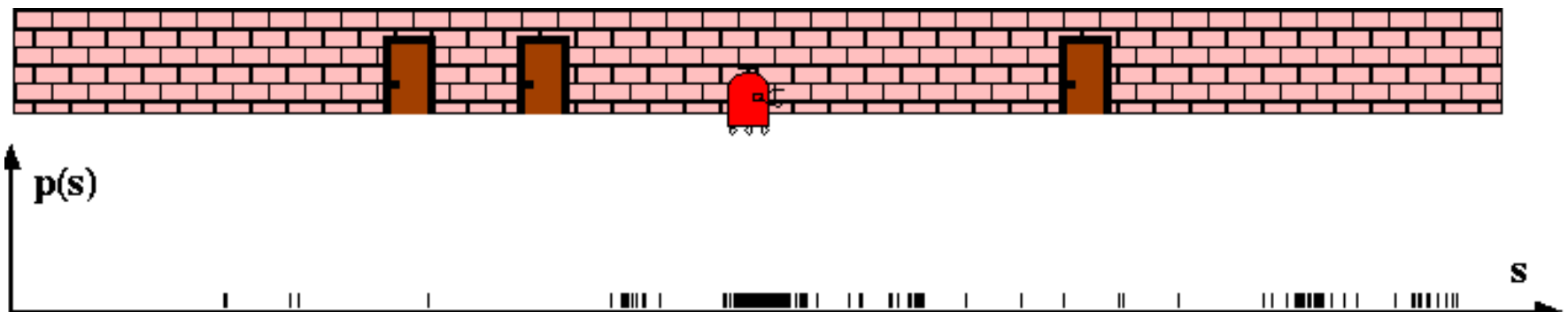
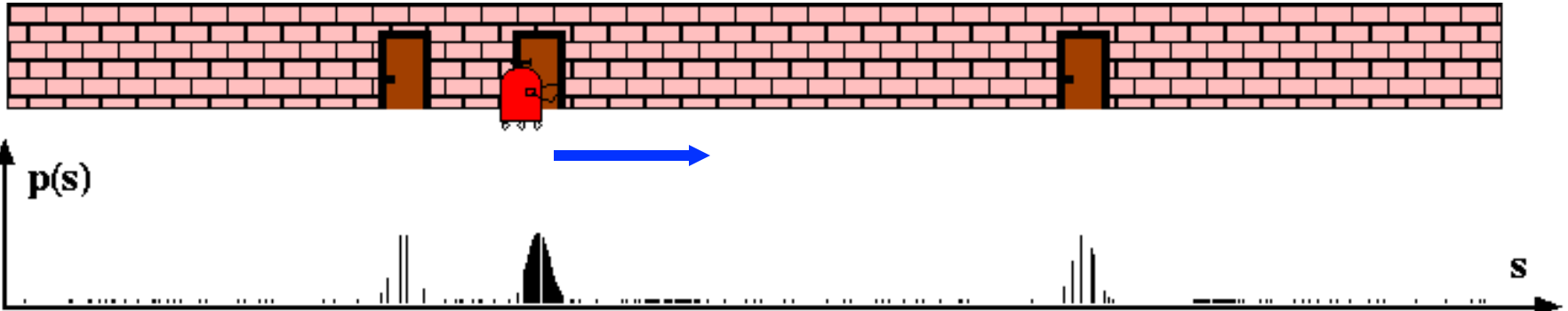
Sensor Information: Importance Sampling

$$\begin{aligned}
 Bel(x) &\leftarrow \alpha p(z|x) Bel^-(x) \\
 w &\leftarrow \frac{\alpha p(z|x) Bel^-(x)}{Bel^-(x)} = \alpha p(z|x)
 \end{aligned}$$



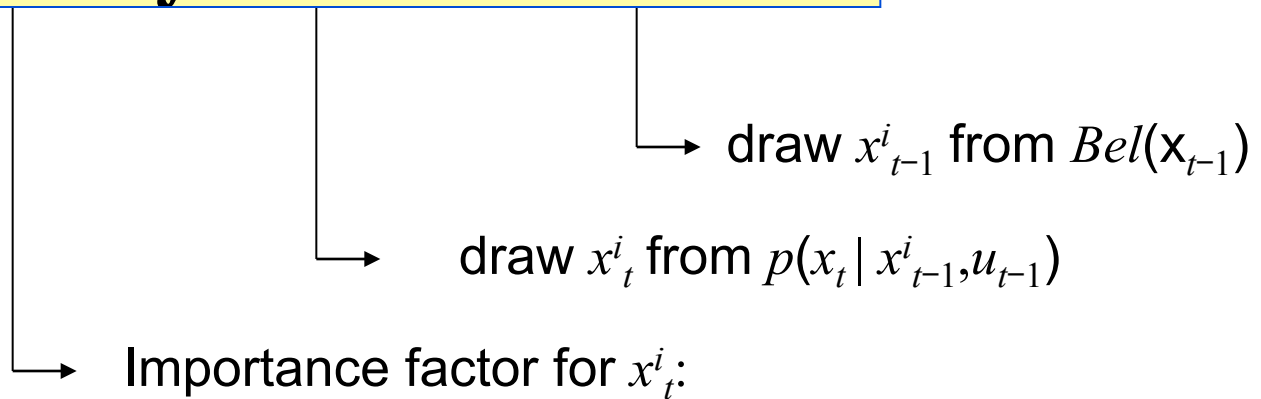
Robot Motion

$$Bel^-(x) \leftarrow \int p(x|u, x') Bel(x') dx'$$



Particle Filter Algorithm

$$Bel(x_t) = \eta p(z_t | x_t) \int p(x_t | x_{t-1}, u_{t-1}) Bel(x_{t-1}) dx_{t-1}$$

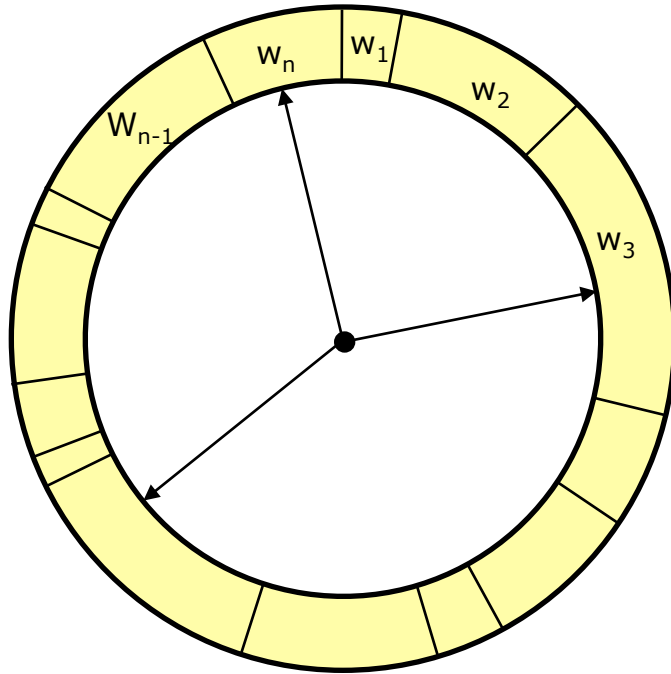


$$\begin{aligned} w_t^i &= \frac{\text{target distribution}}{\text{proposal distribution}} \\ &= \frac{\eta p(z_t | x_t) p(x_t | x_{t-1}, u_{t-1}) Bel(x_{t-1})}{p(x_t | x_{t-1}, u_{t-1}) Bel(x_{t-1})} \\ &\propto p(z_t | x_t) \end{aligned}$$

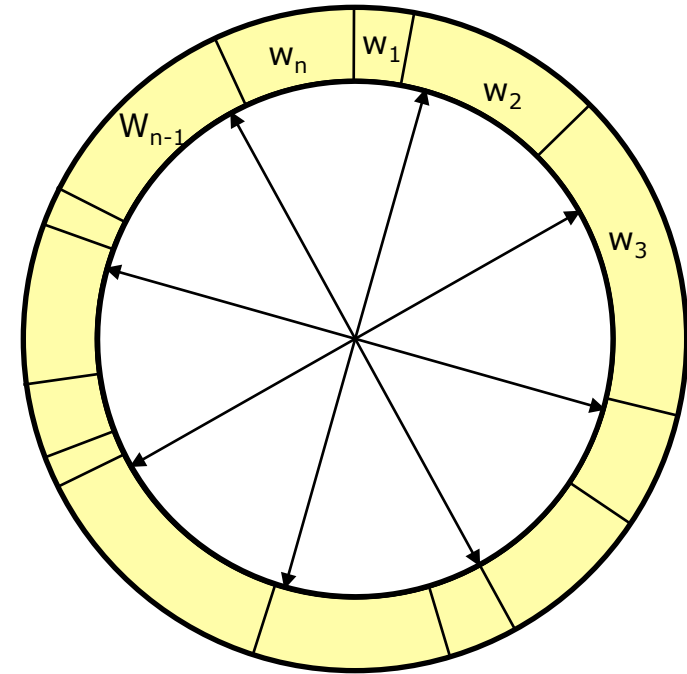
Resampling

- **Given**: Set S of weighted samples.
- **Wanted** : Random sample, where the probability of drawing x_i is given by w_i .
- Typically done n times with replacement to generate new sample set S' .

Resampling

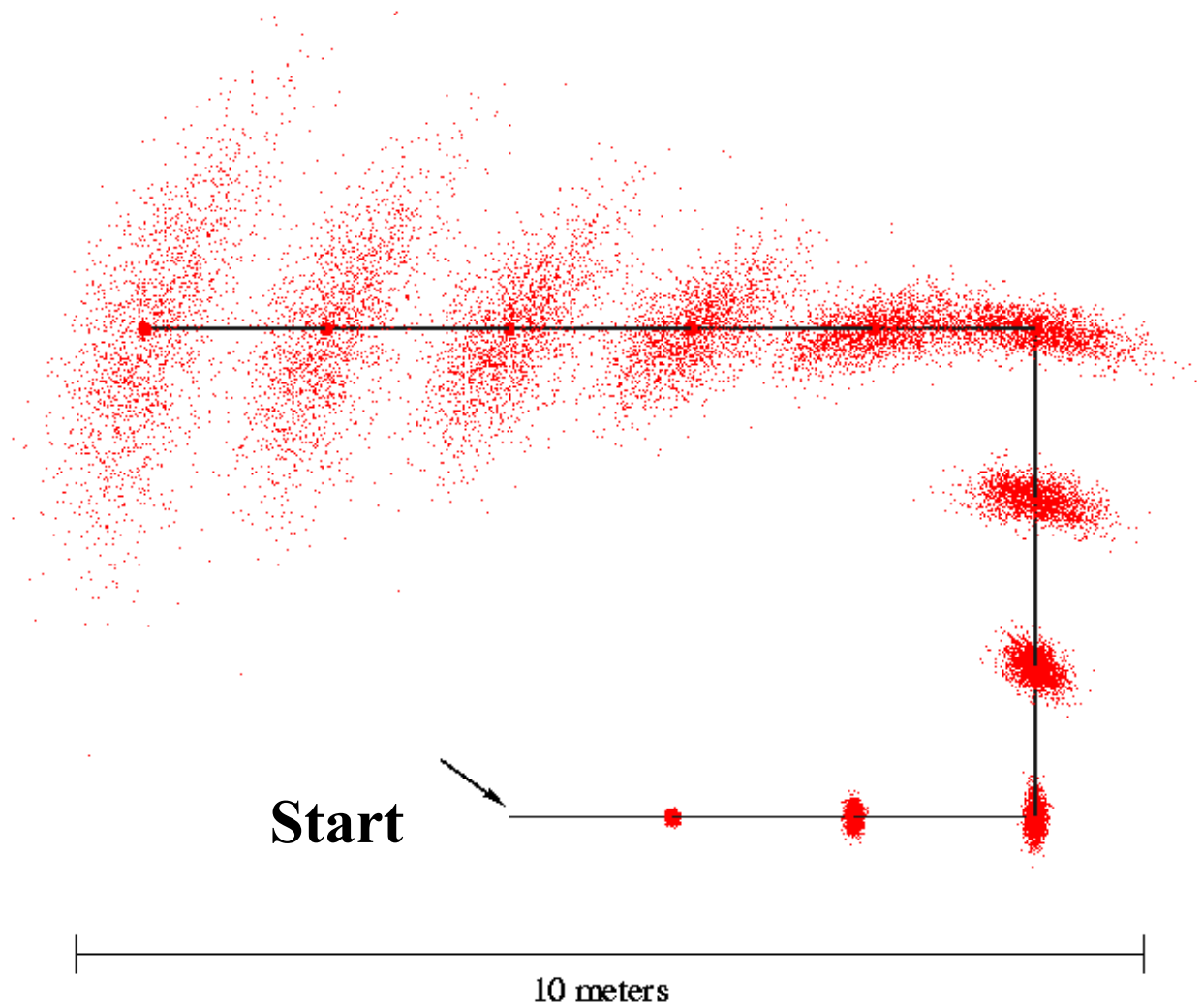


- Roulette wheel
- Binary search, $n \log n$

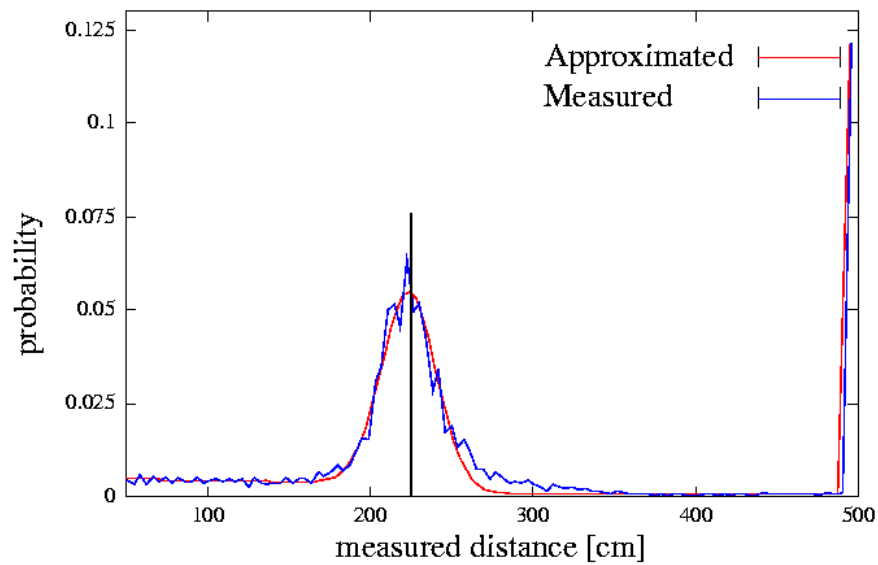


- Stochastic universal sampling
- Systematic resampling
- Linear time complexity
- Easy to implement, low variance

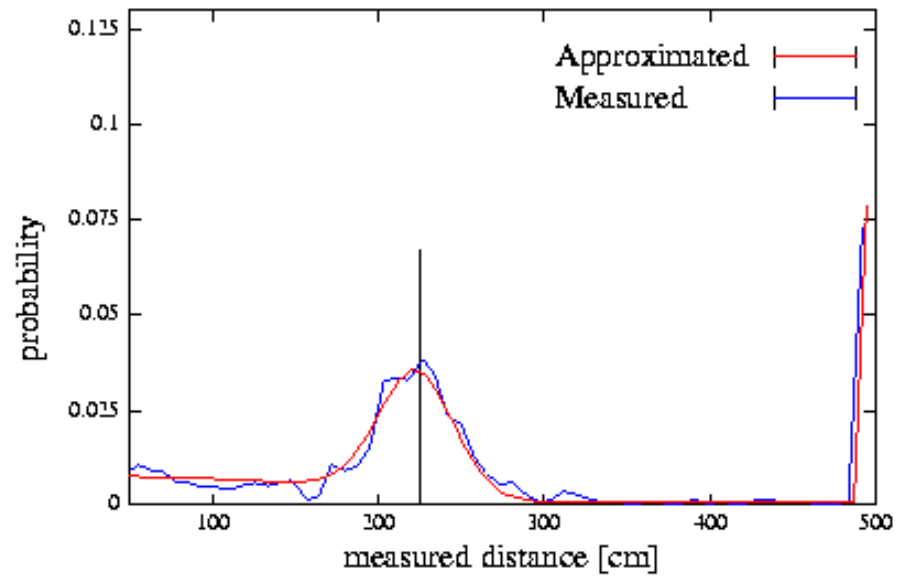
Motion Model Reminder



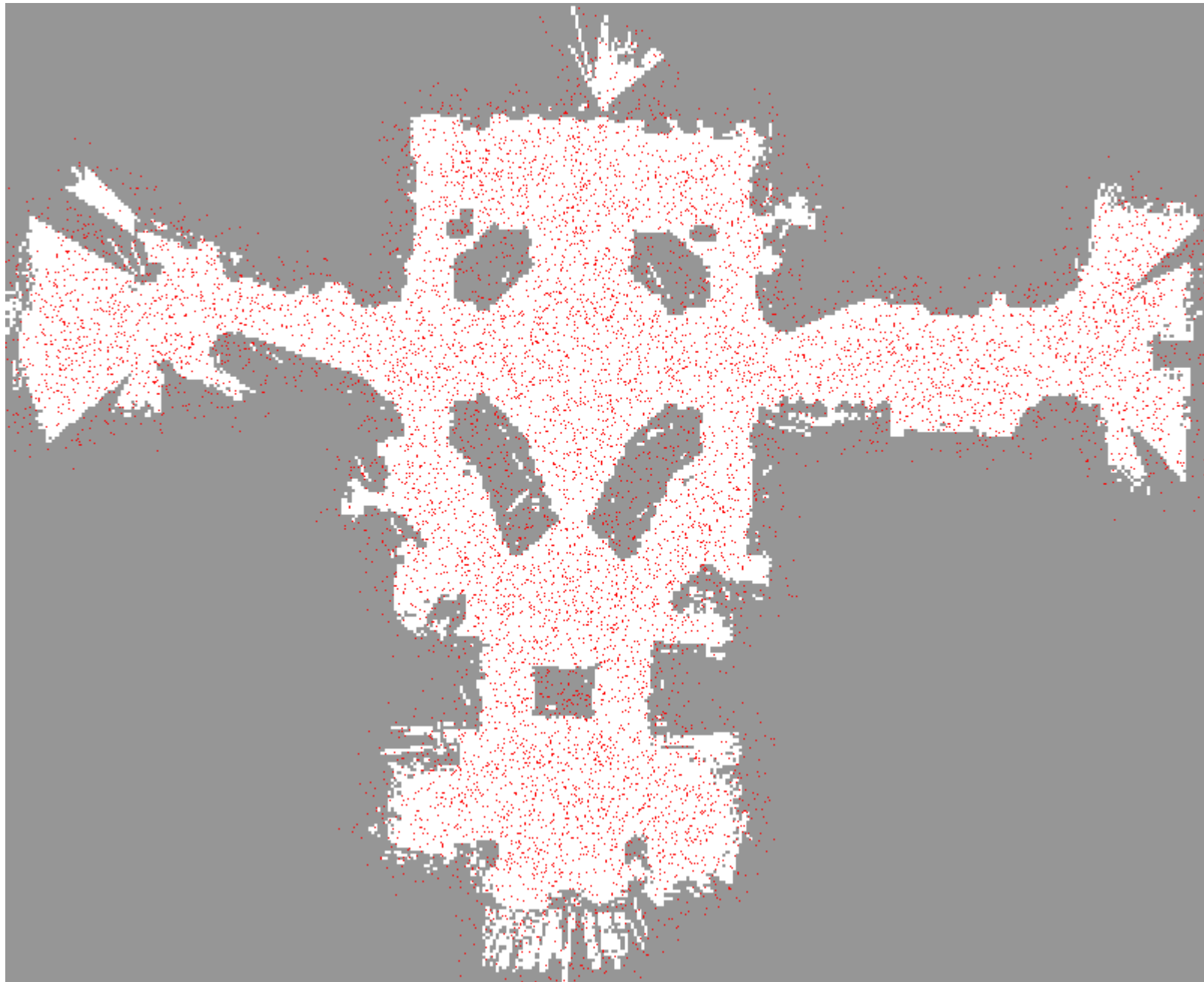
Proximity Sensor Model Reminder

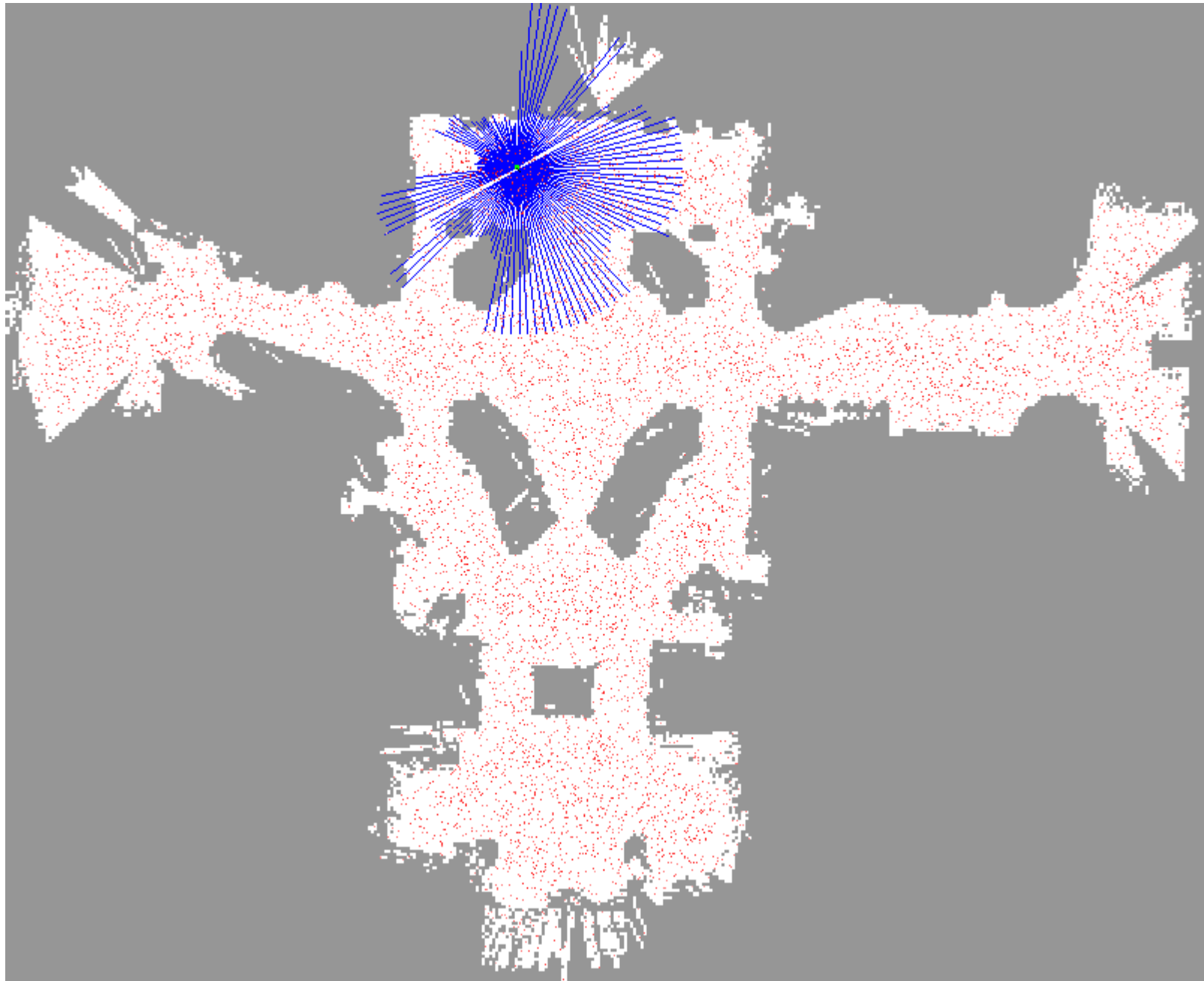


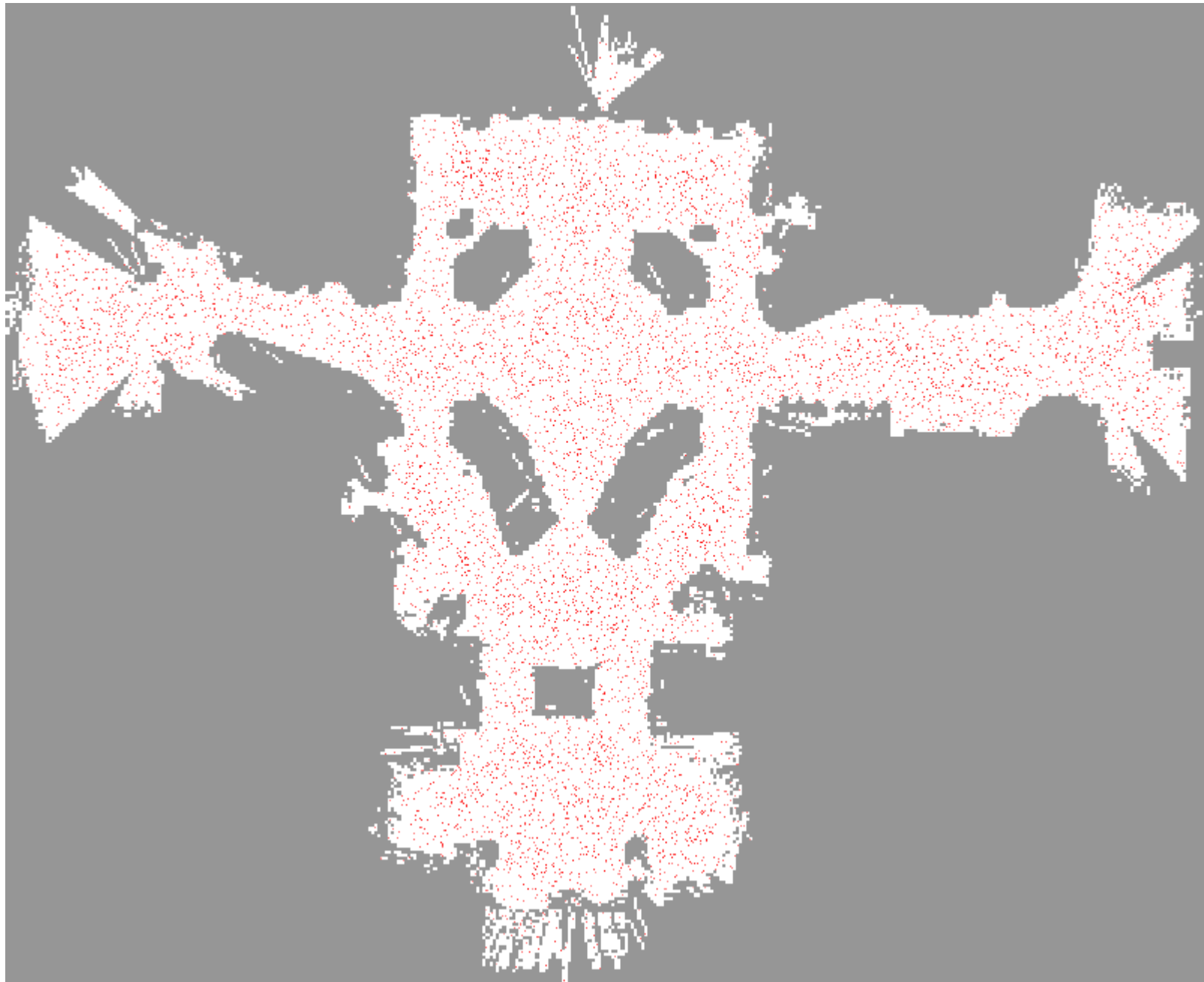
Laser sensor

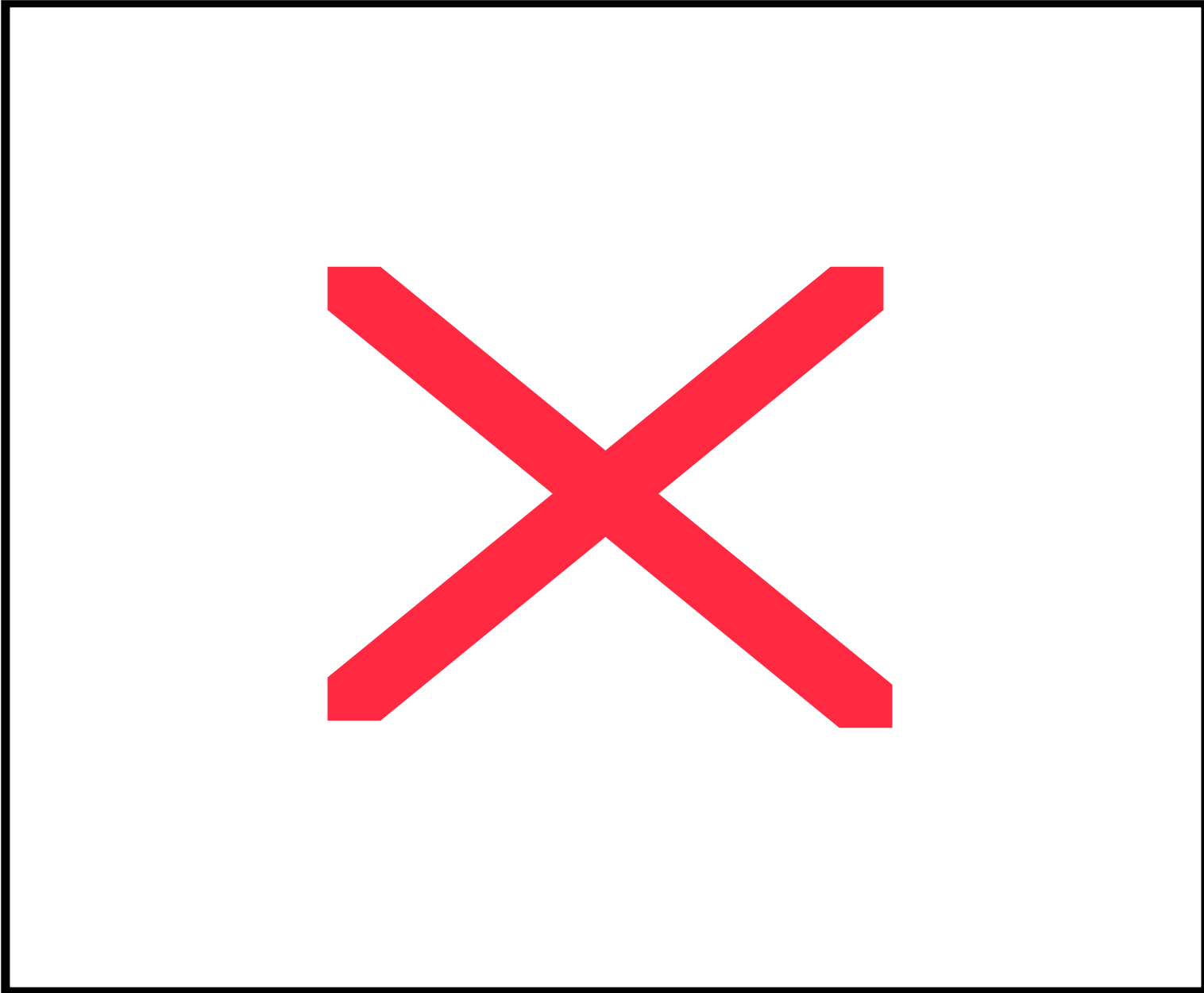


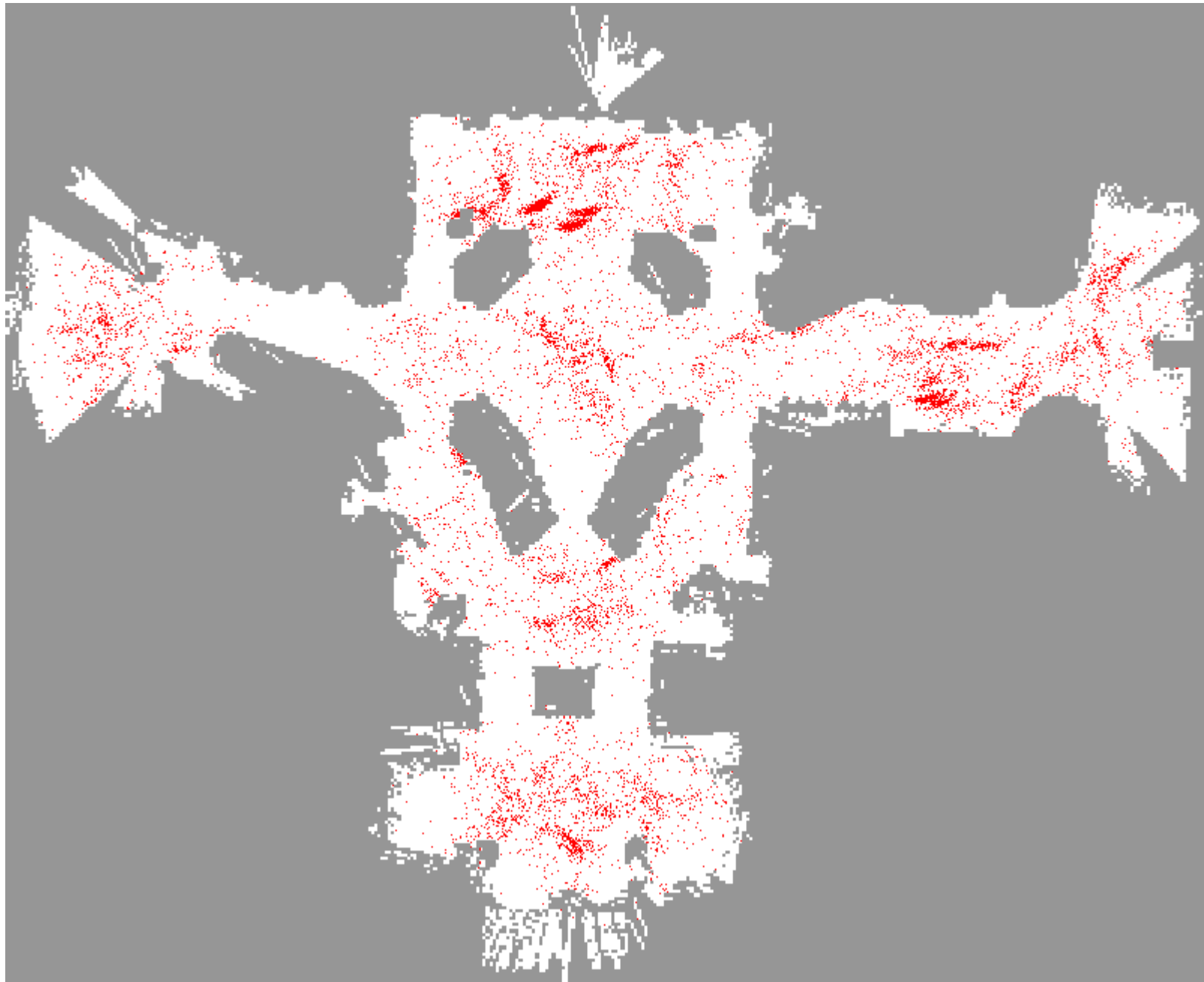
Sonar sensor

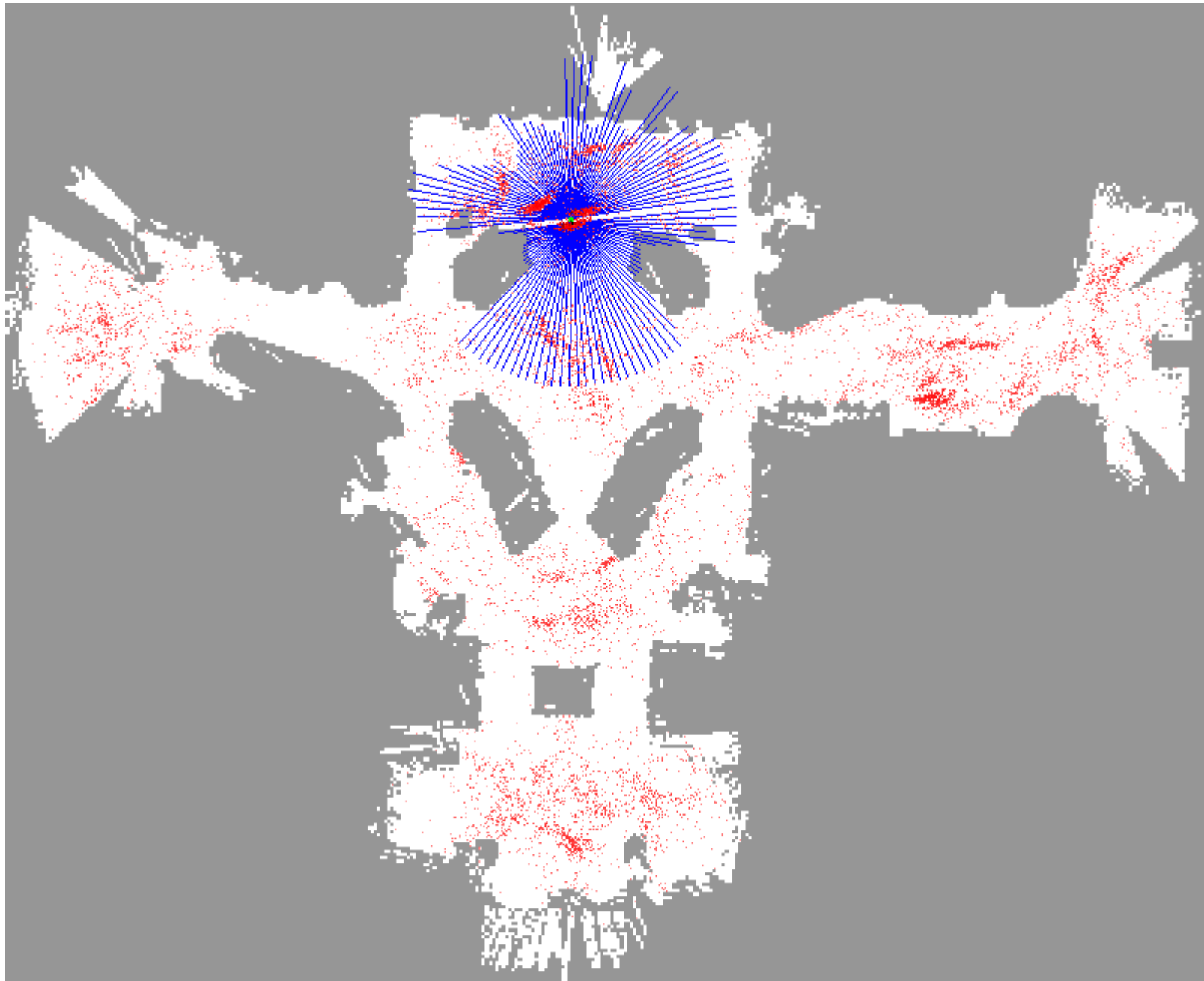


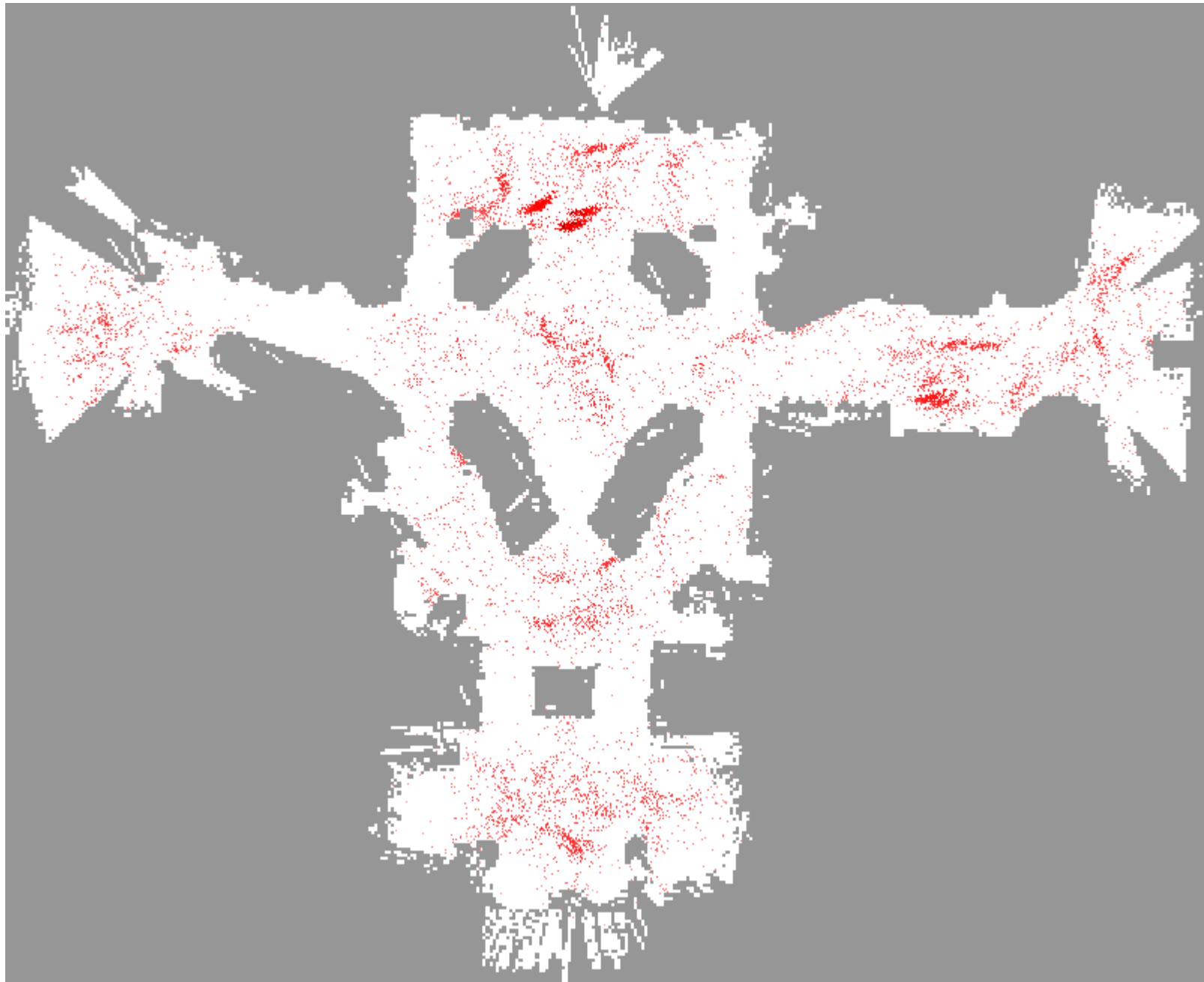


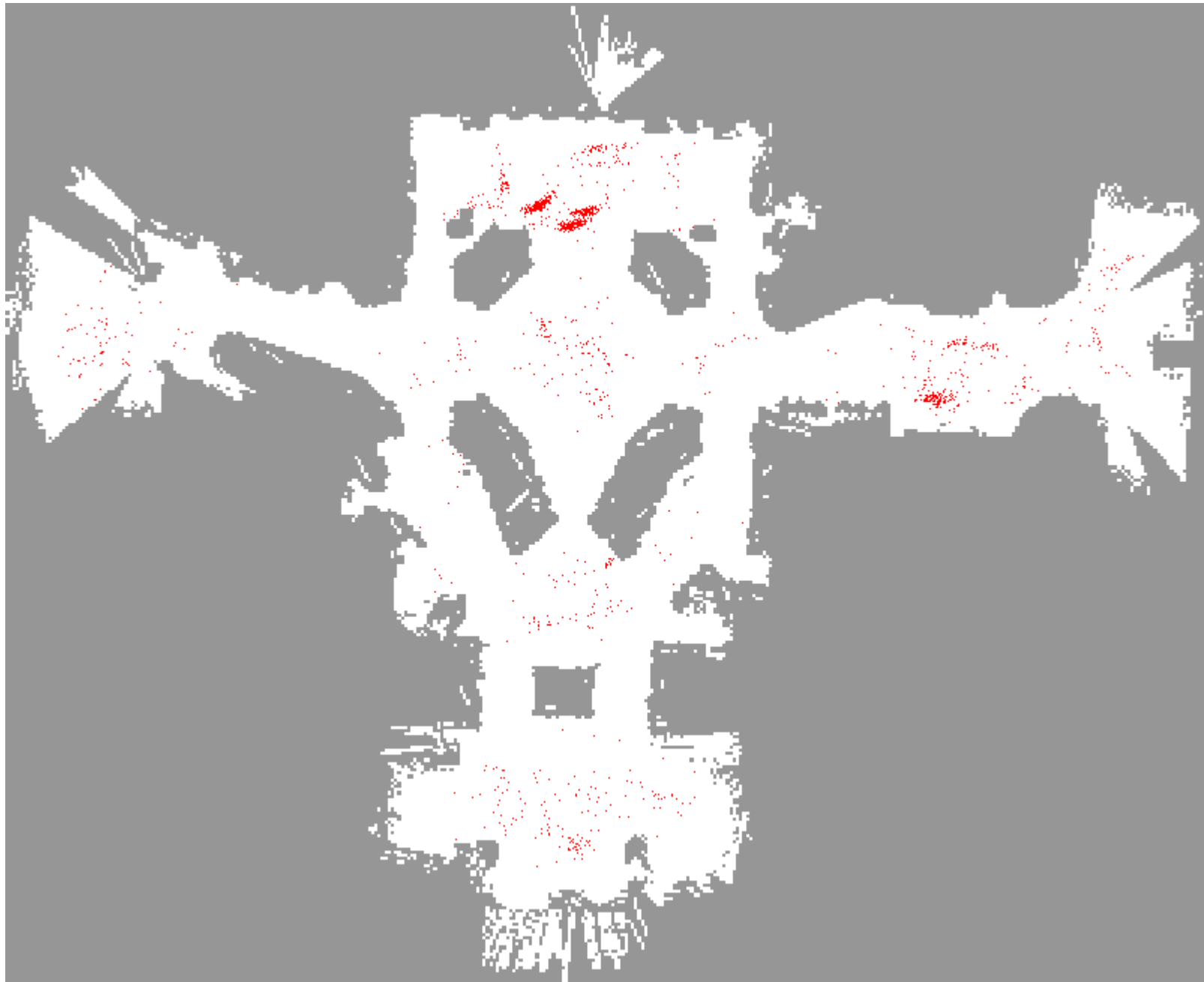




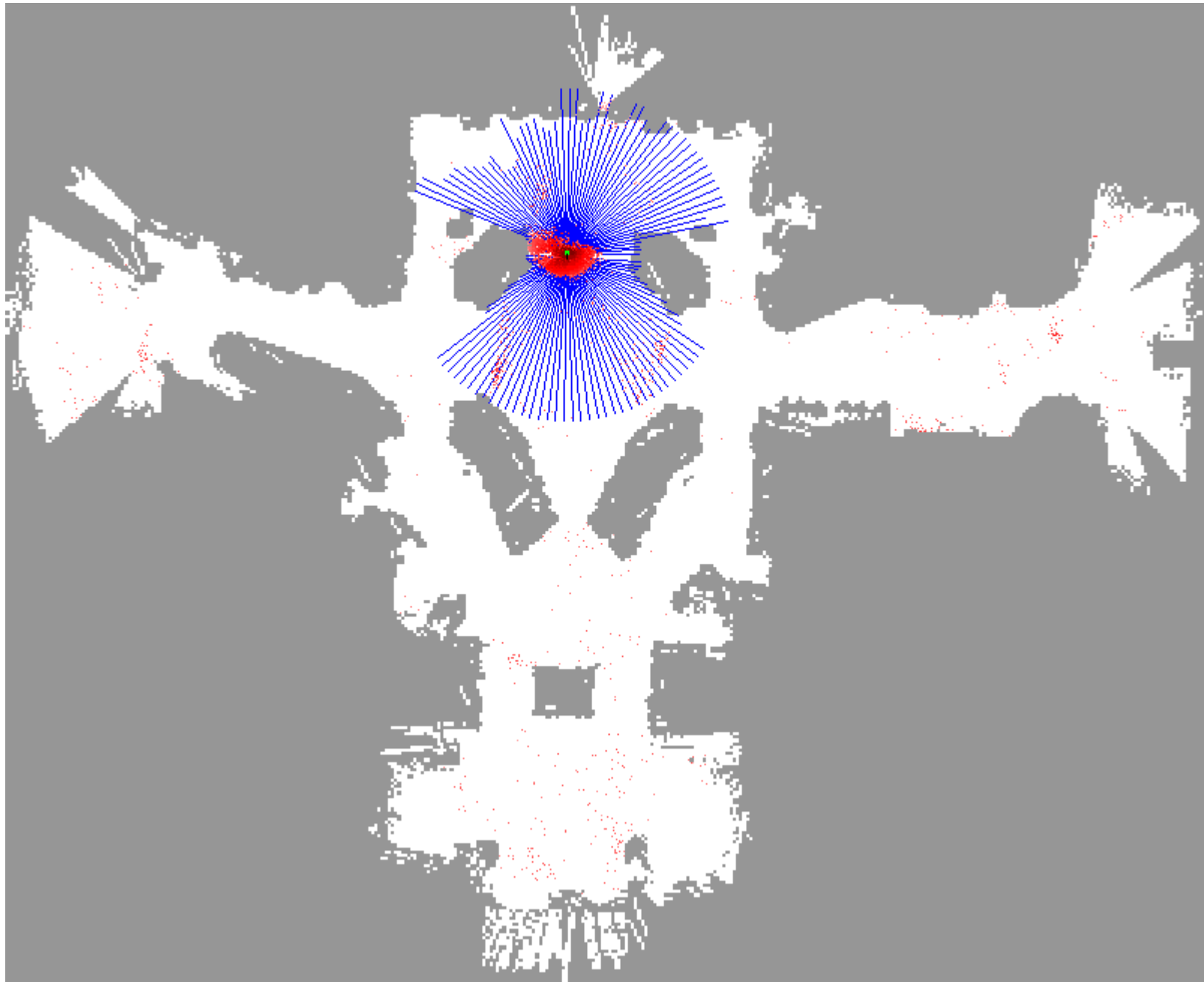




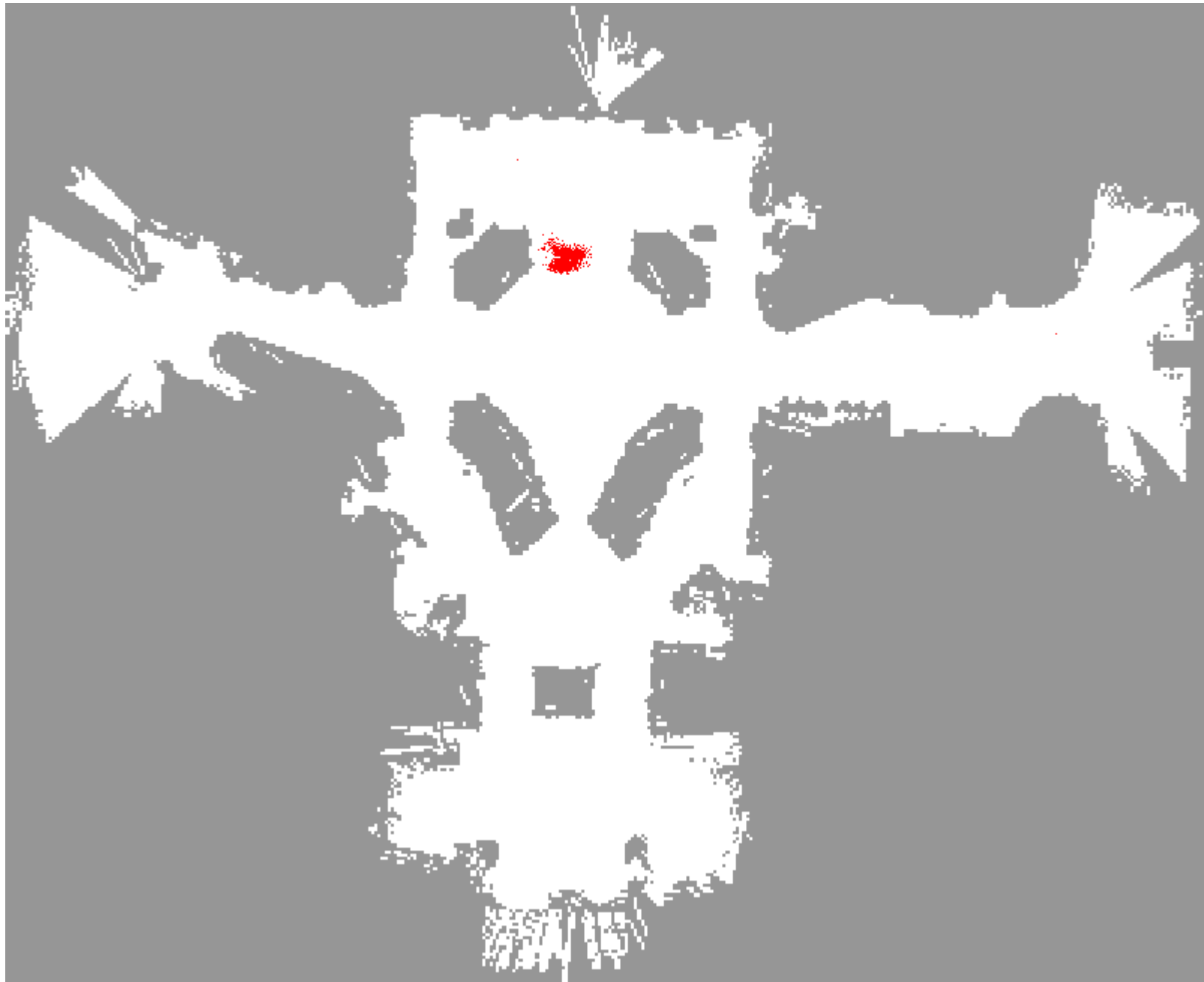


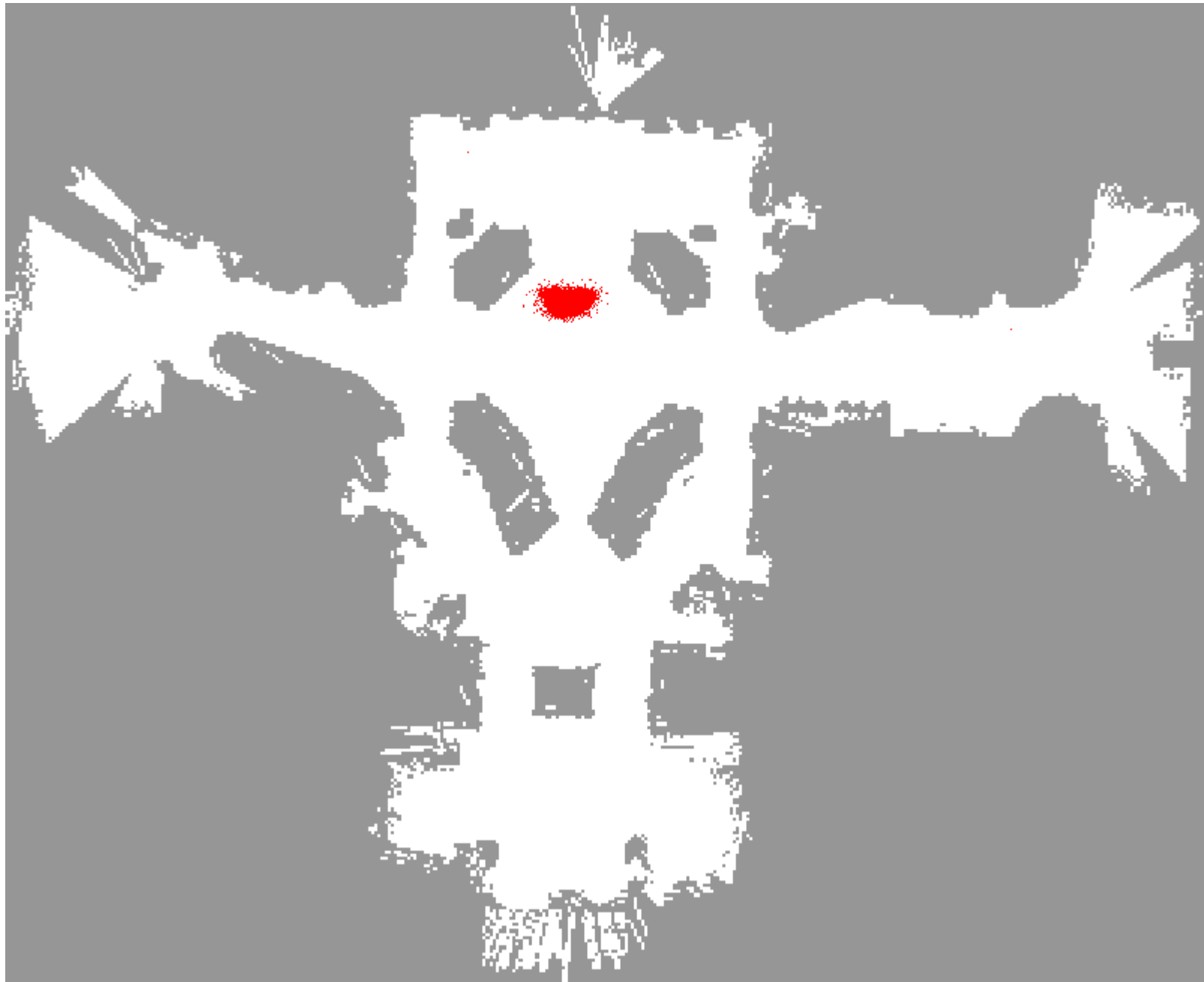


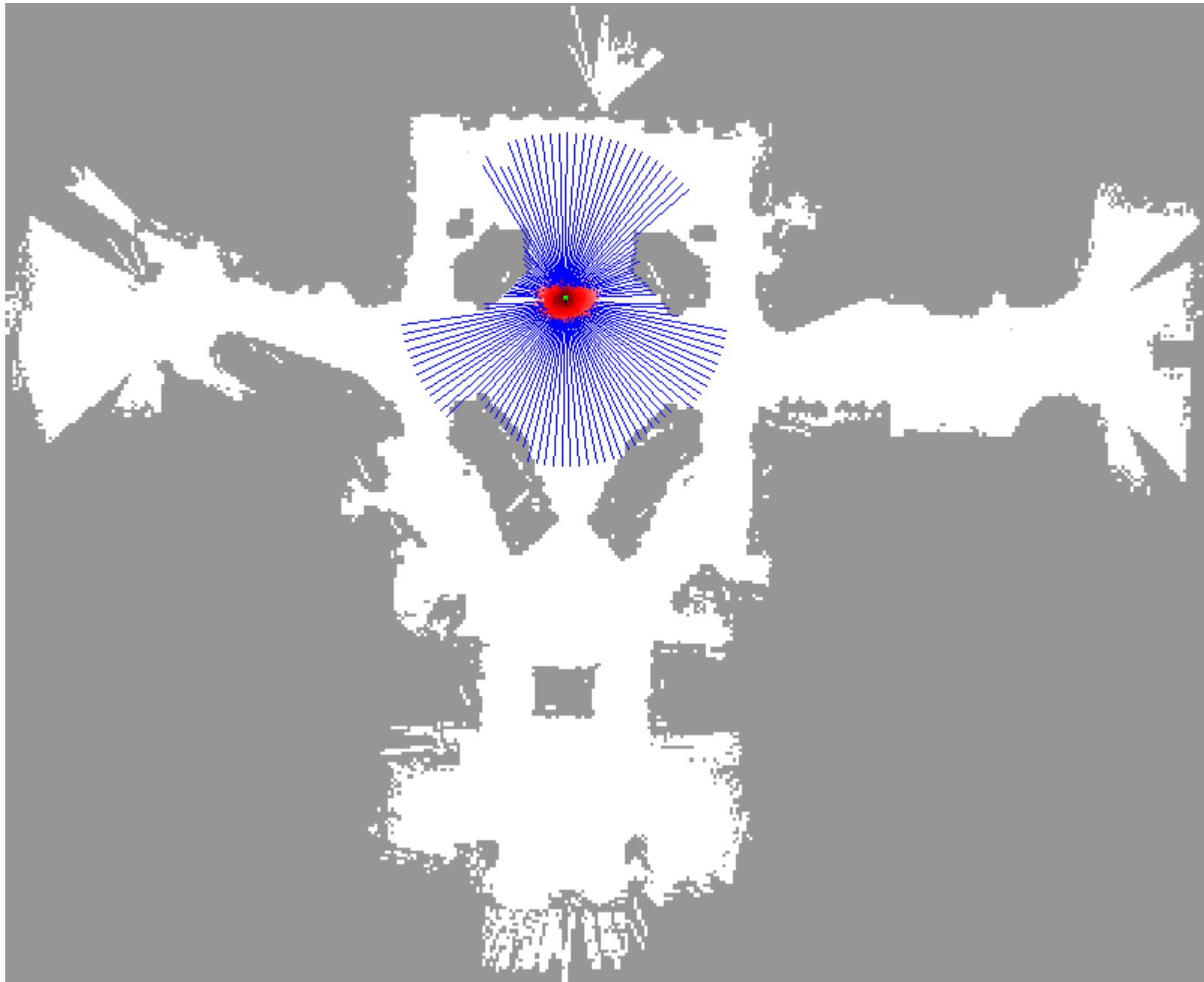


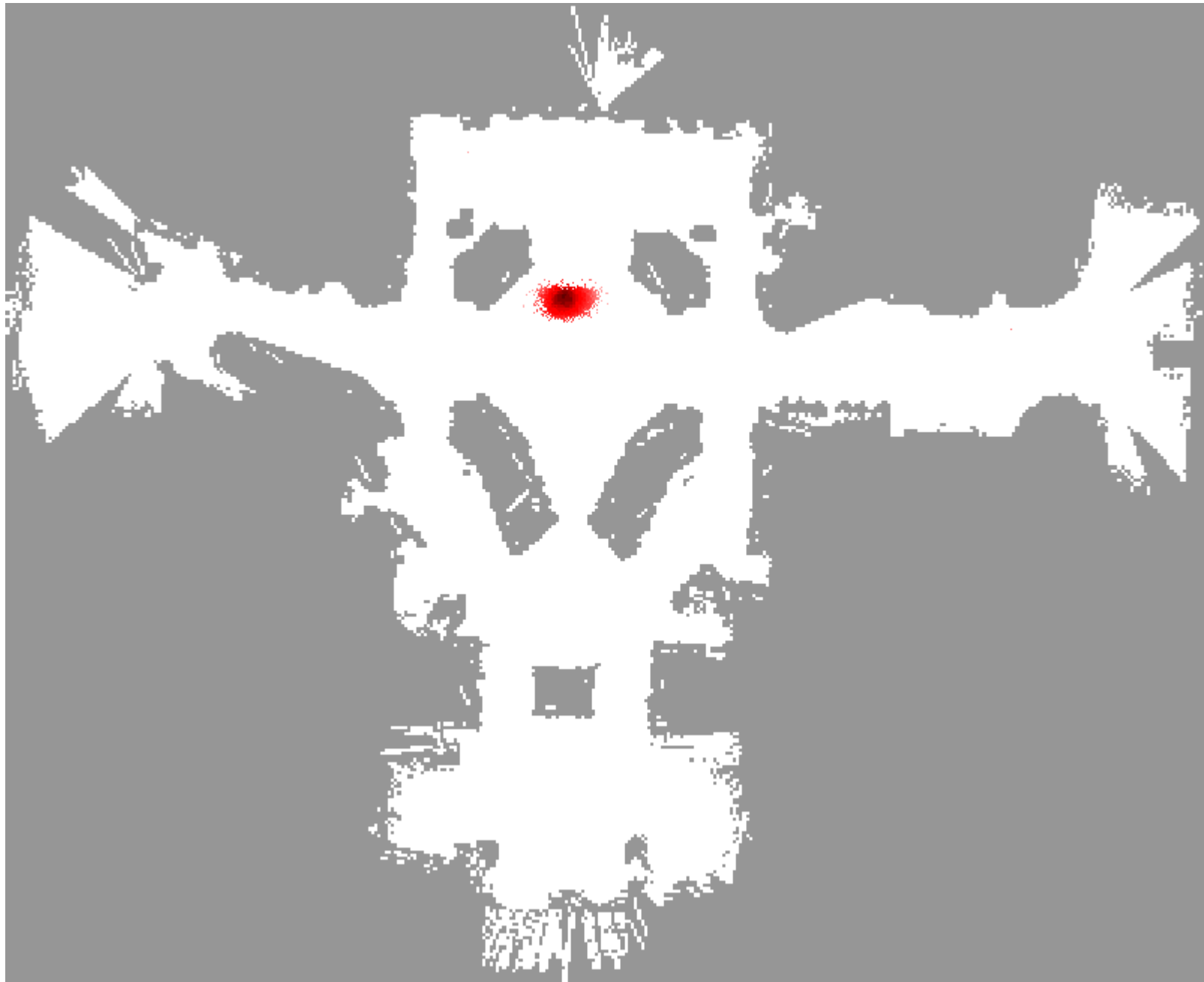


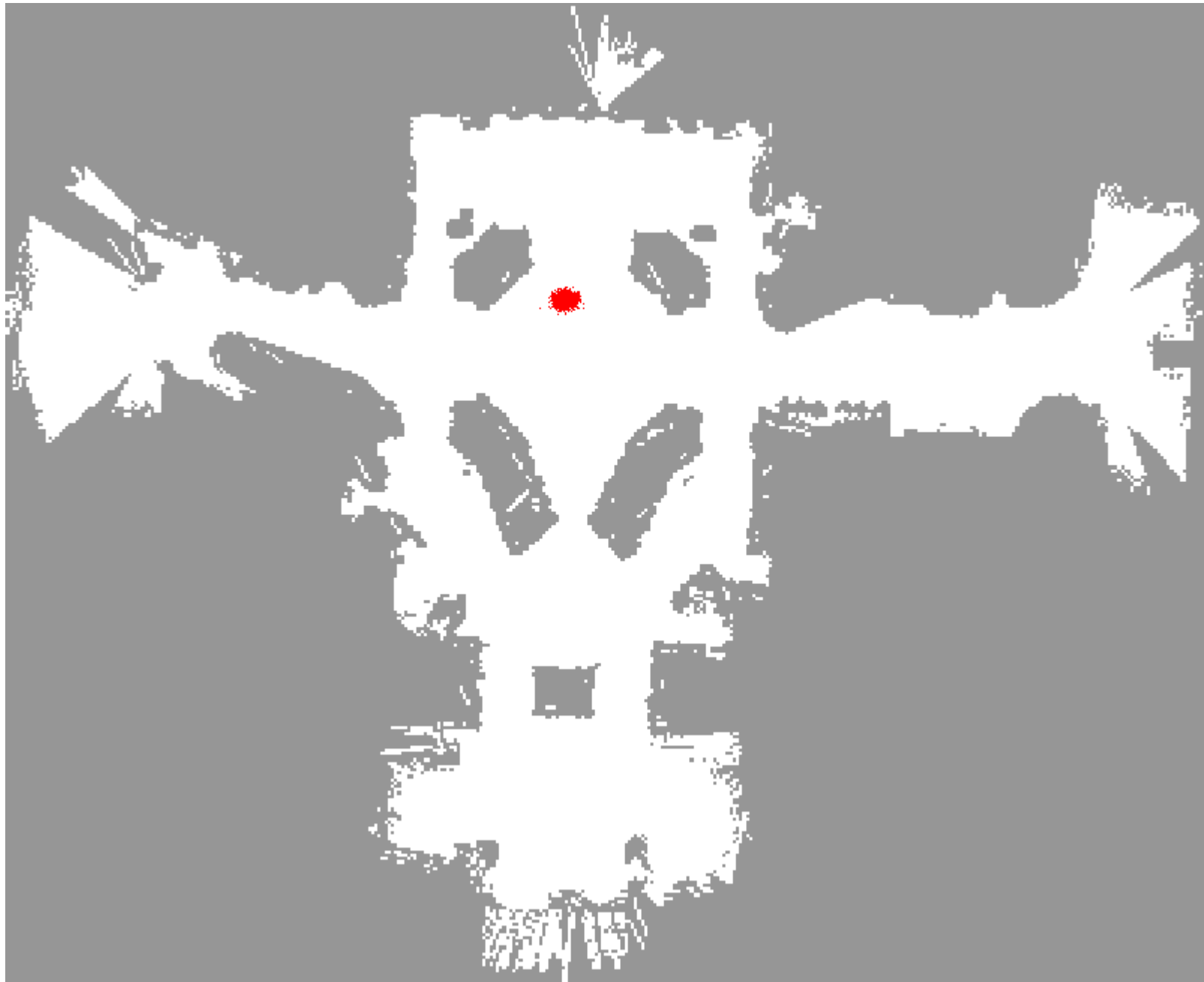


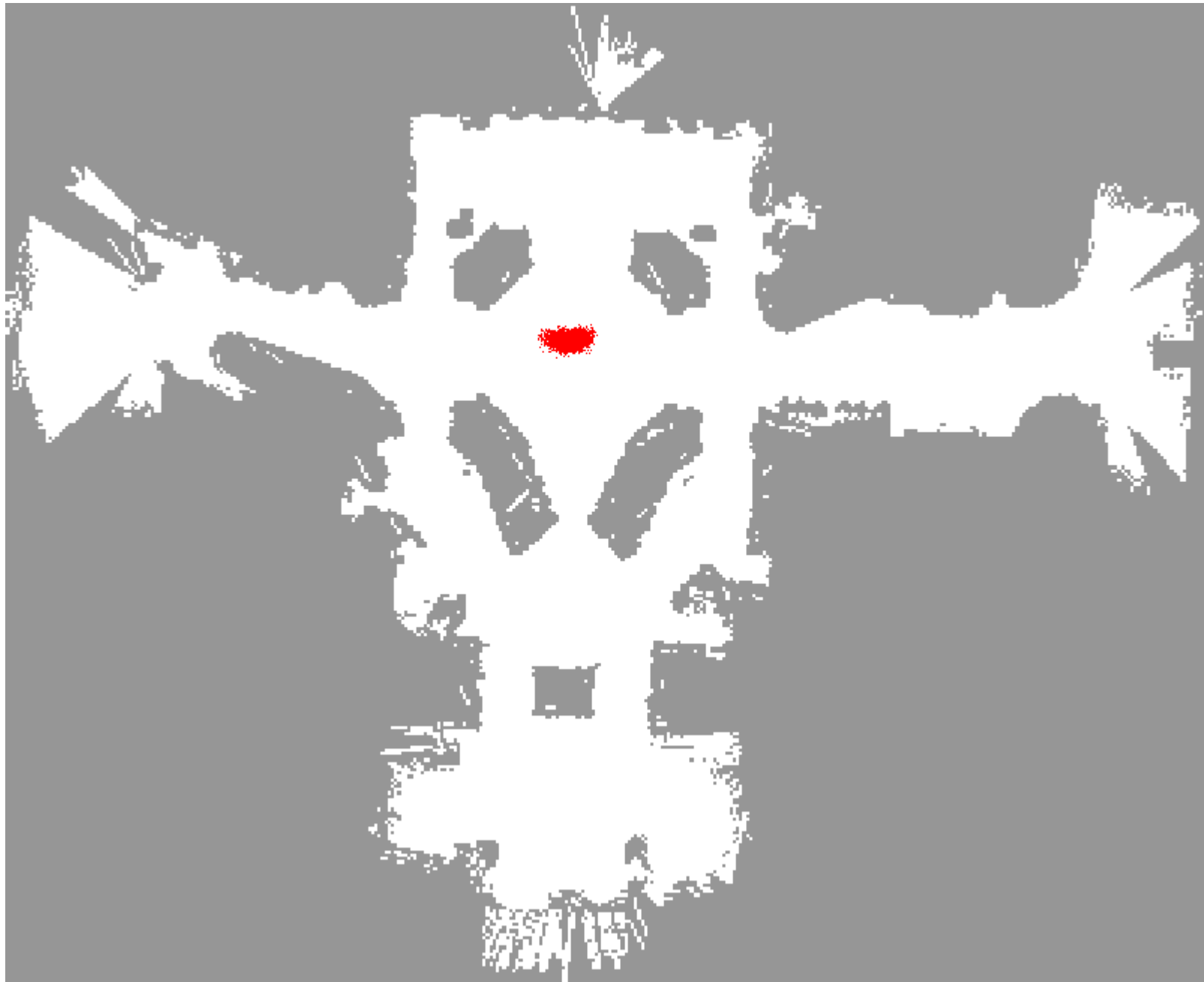


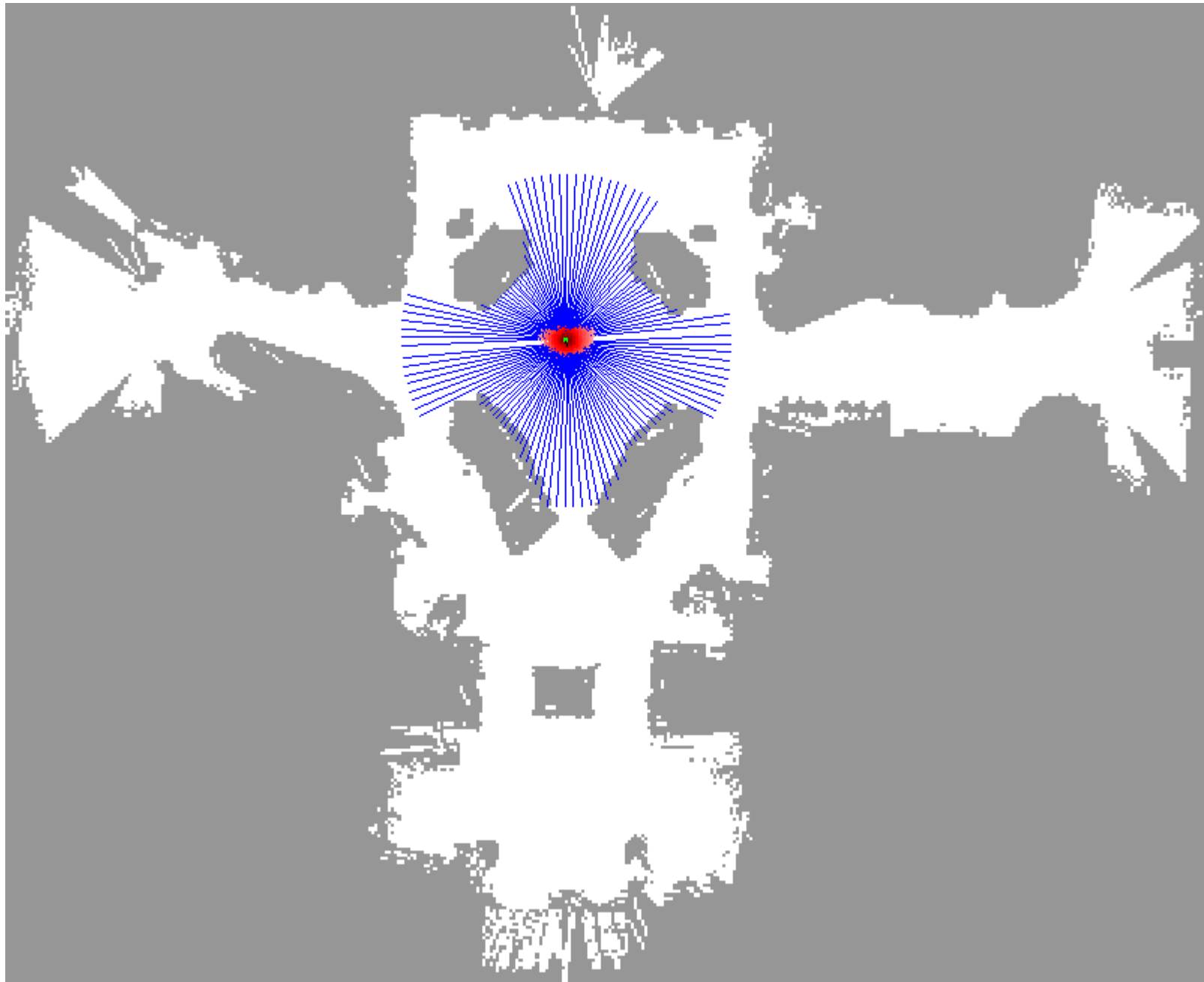


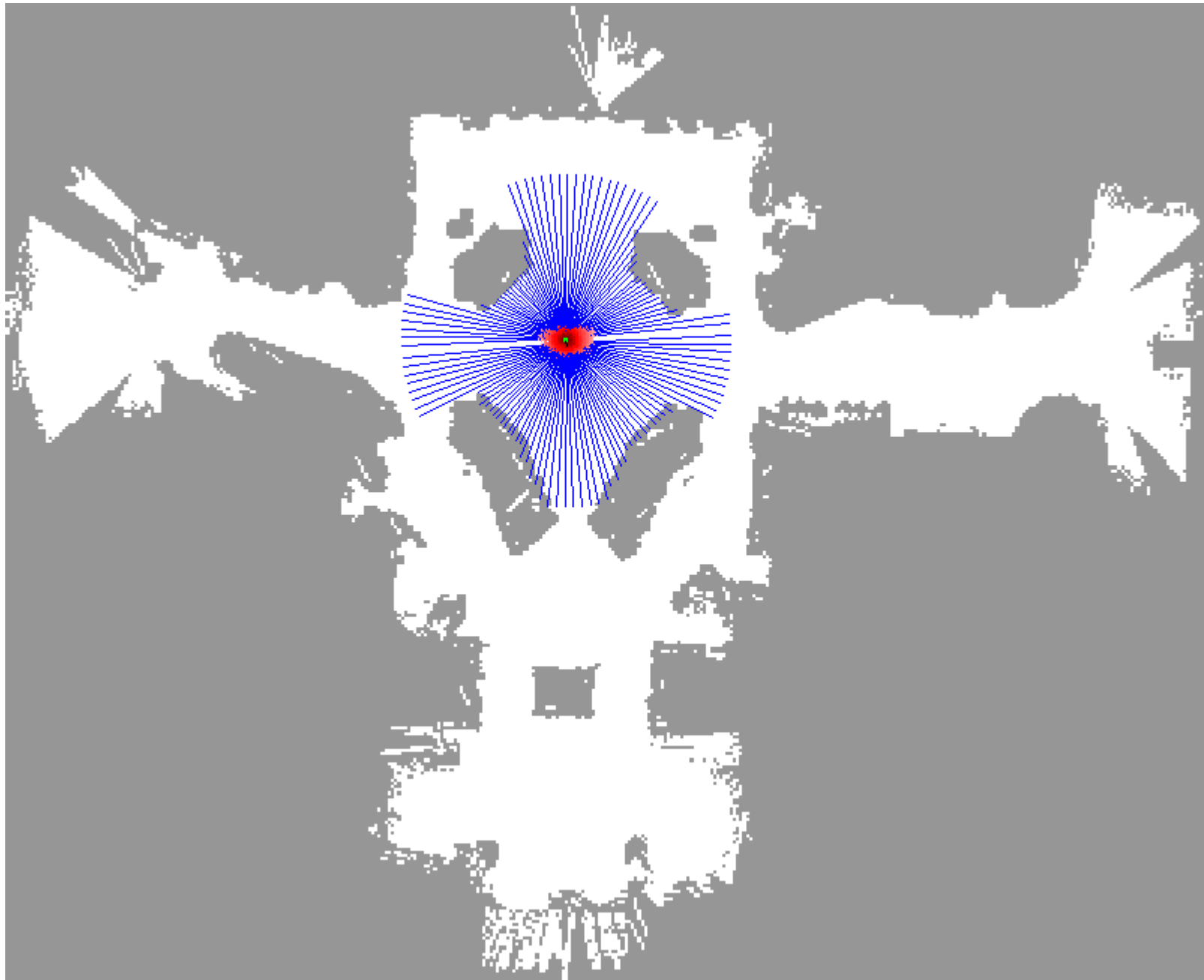




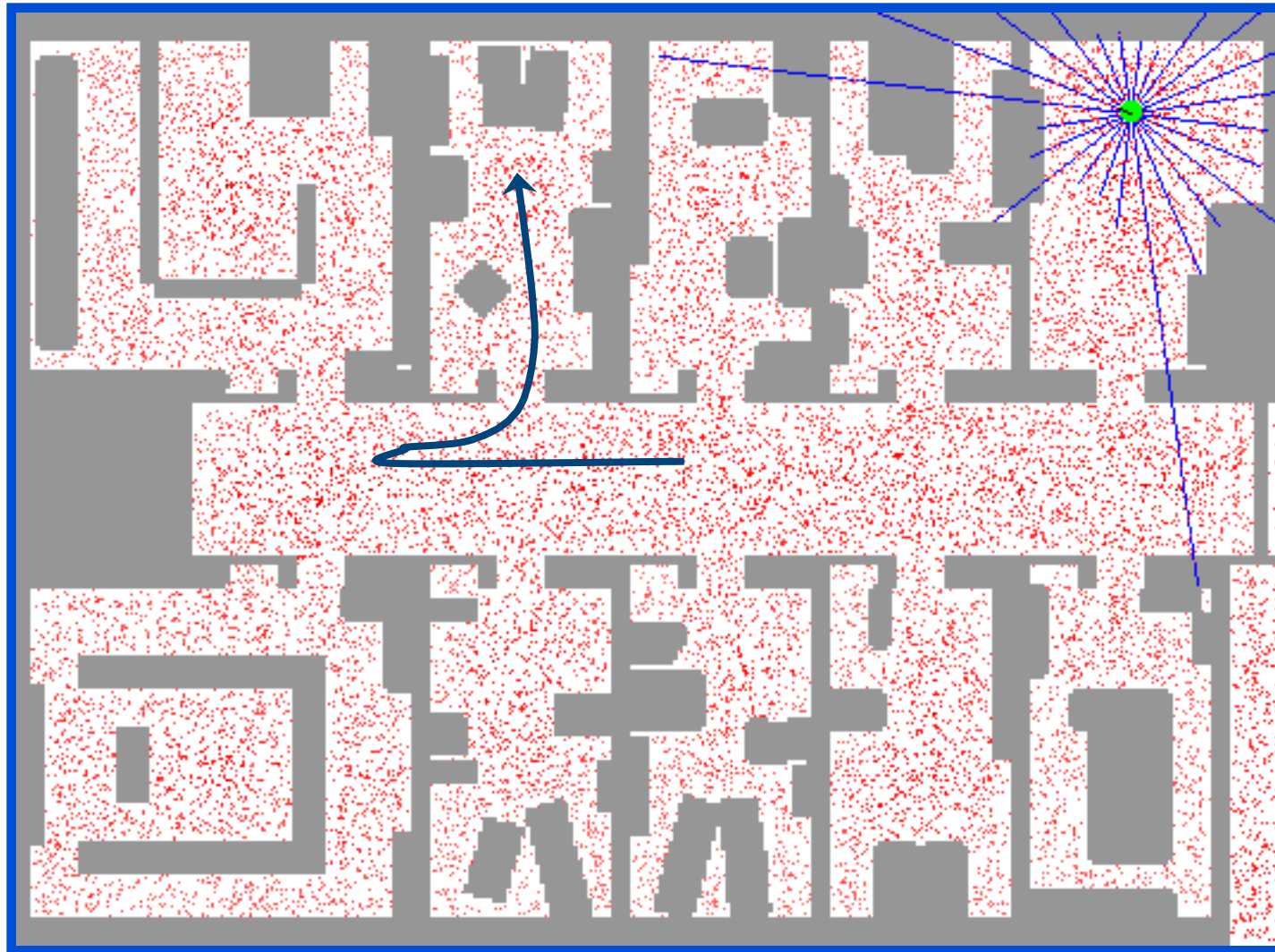




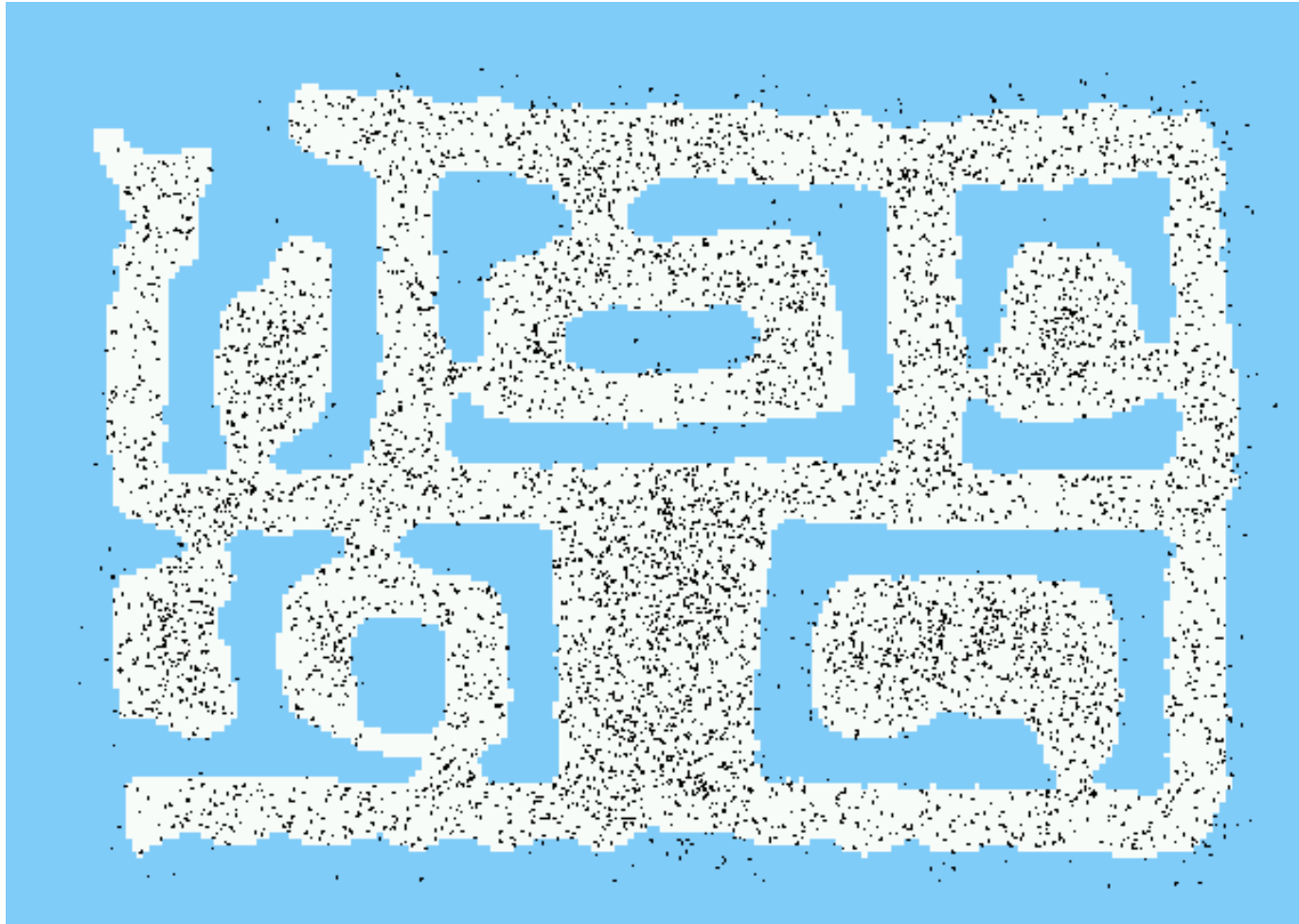




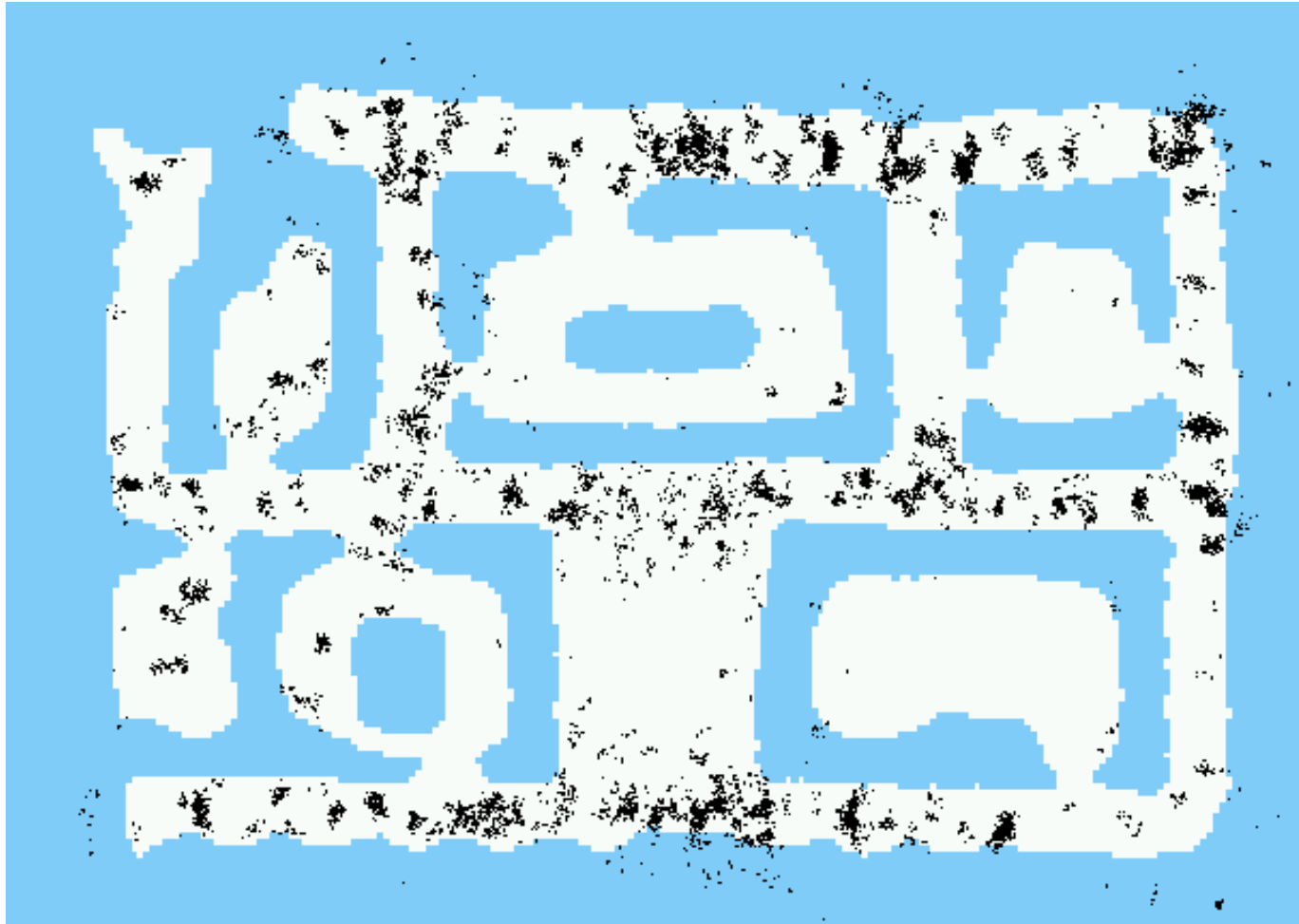
Sample-based Localization (sonar)



Initial Distribution



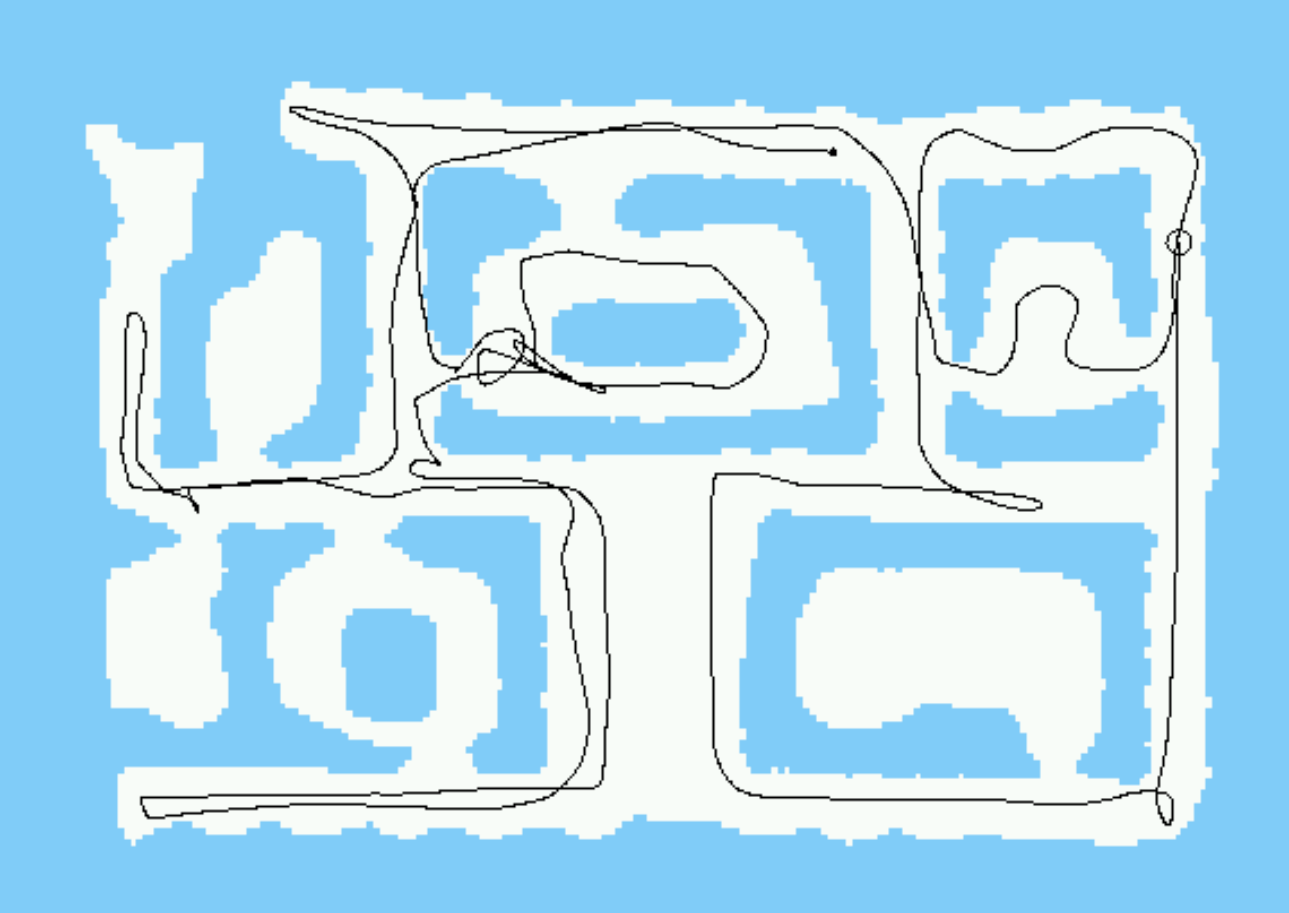
After Incorporating Ten Ultrasound Scans



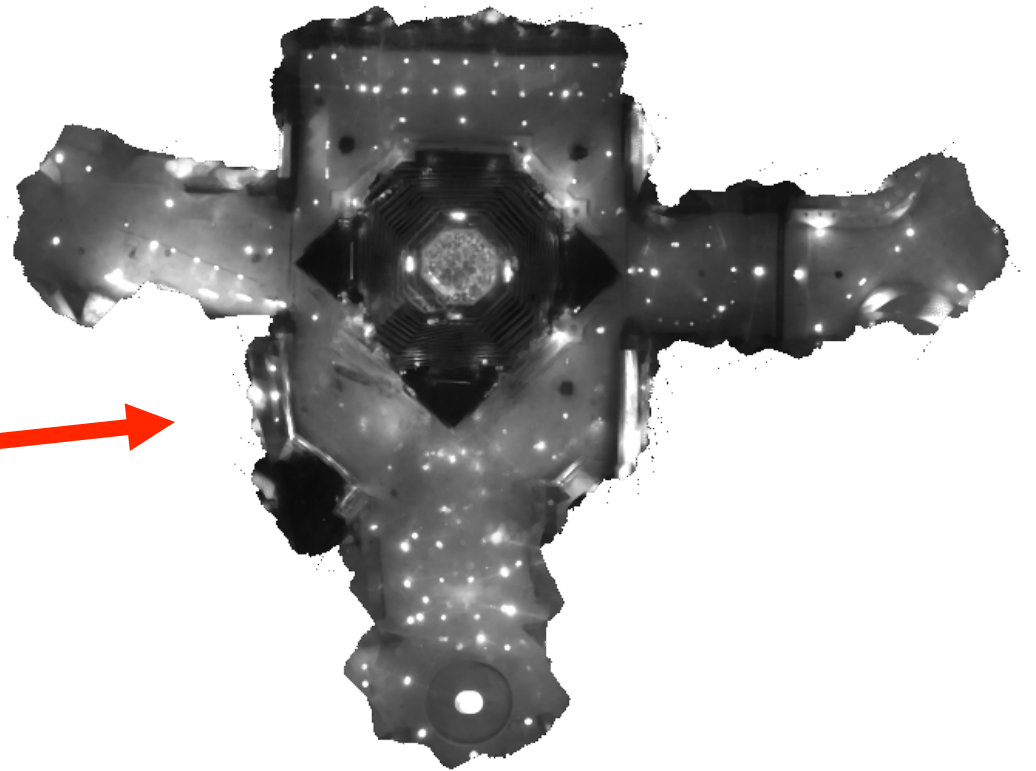
After Incorporating 65 Ultrasound Scans



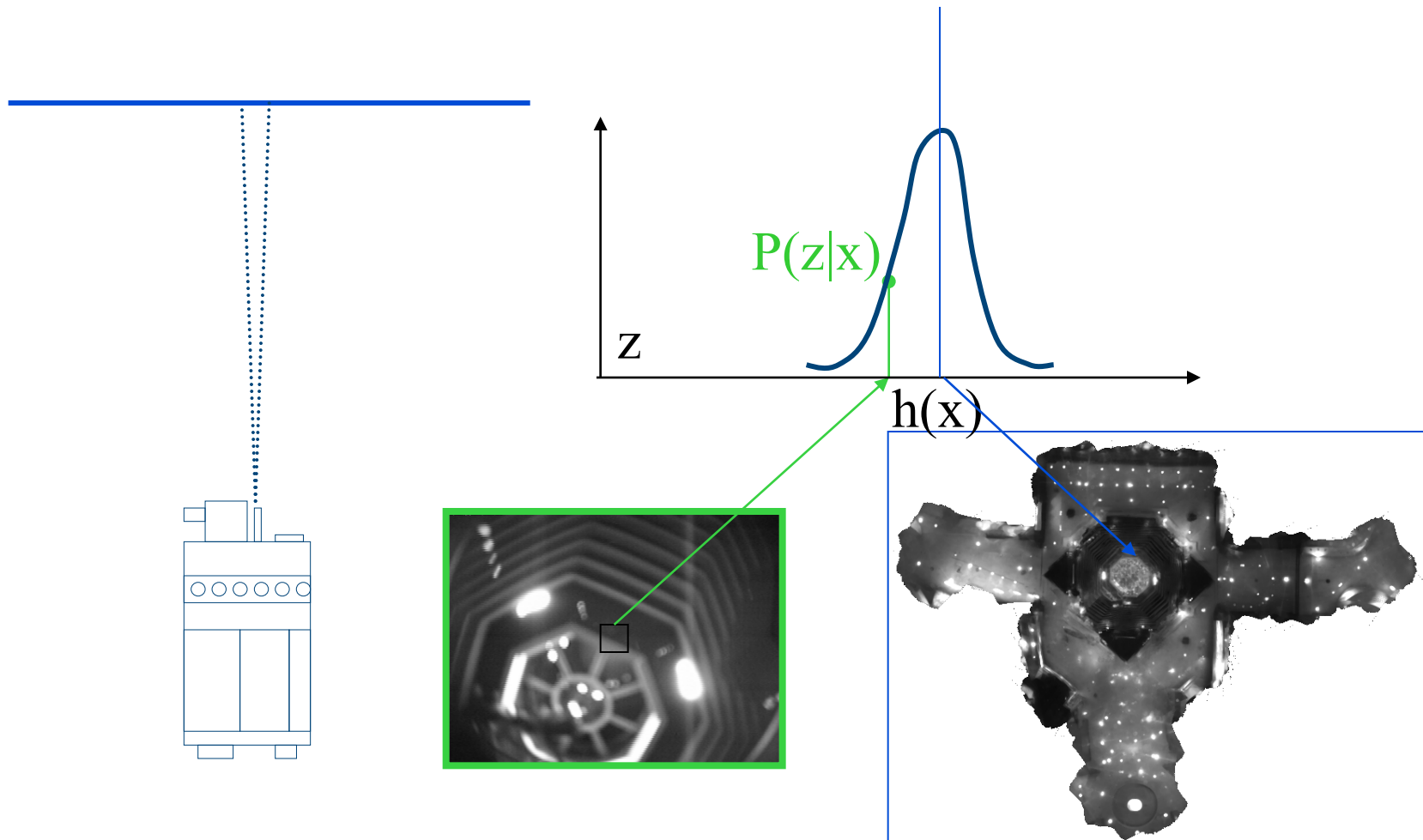
Estimated Path



Using Ceiling Maps for Localization

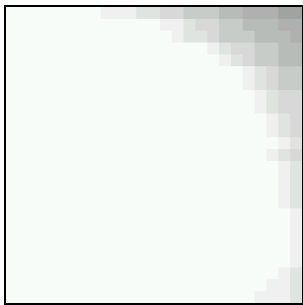


Vision-based Localization

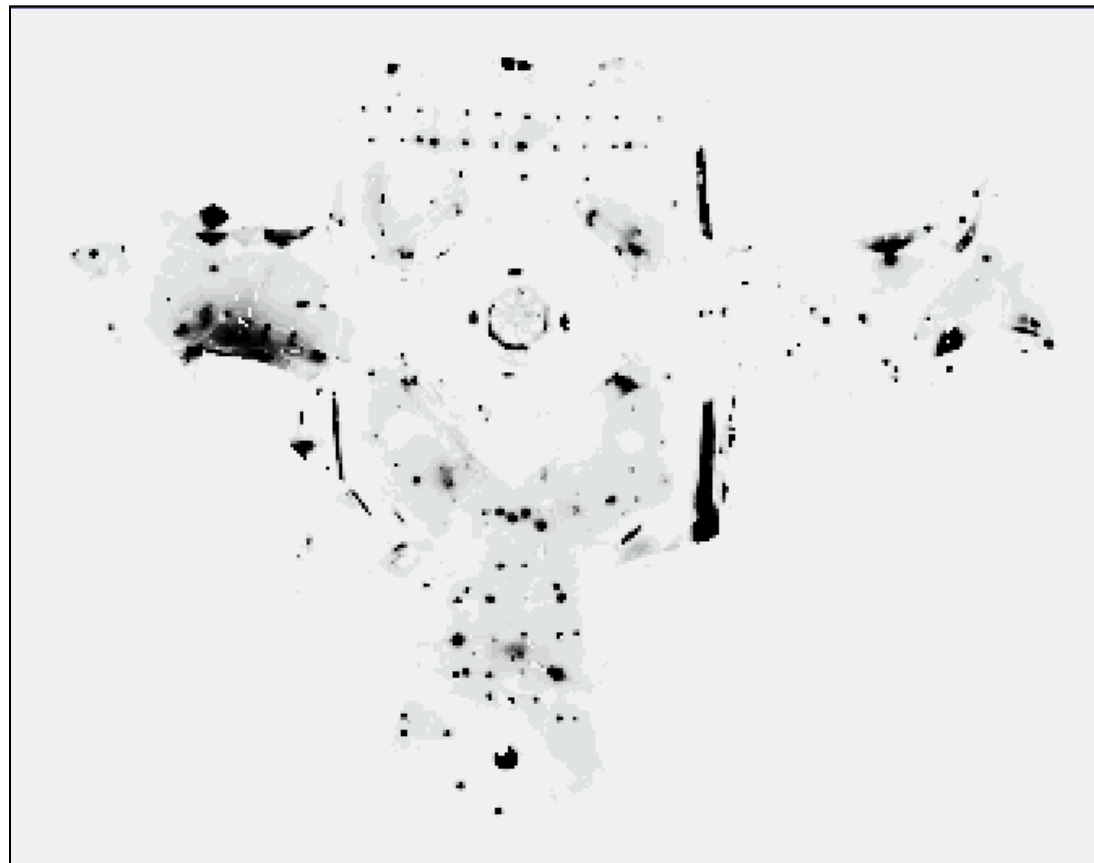


Under a Light

Measurement z :

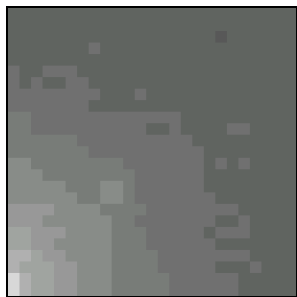


$P(z|x)$:

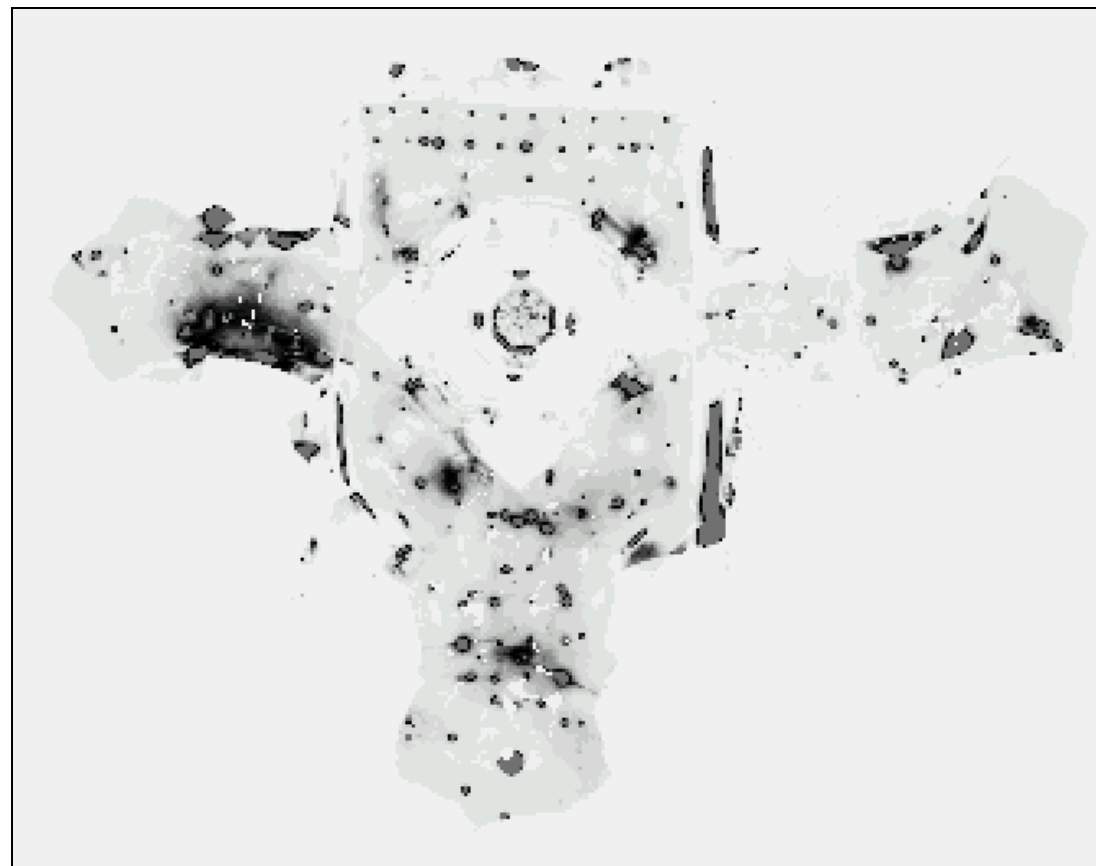


Next to a Light

Measurement z :



$P(z|x)$:

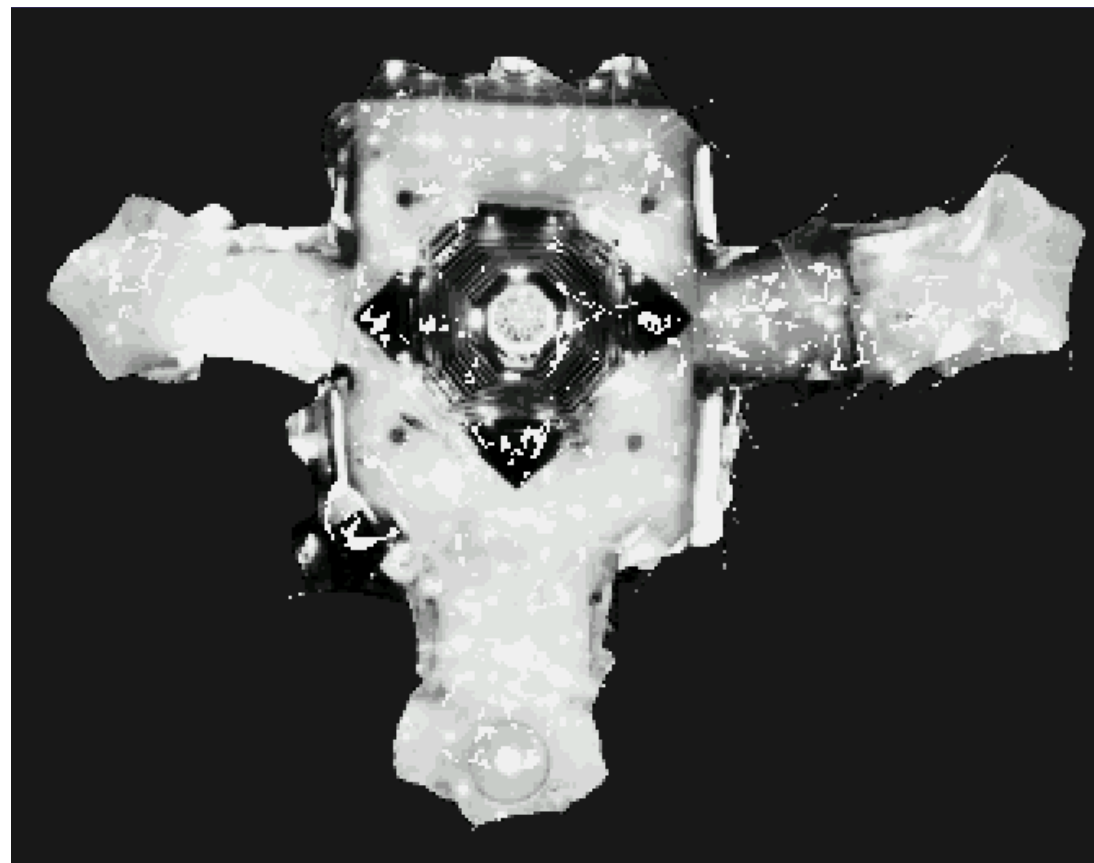


Elsewhere

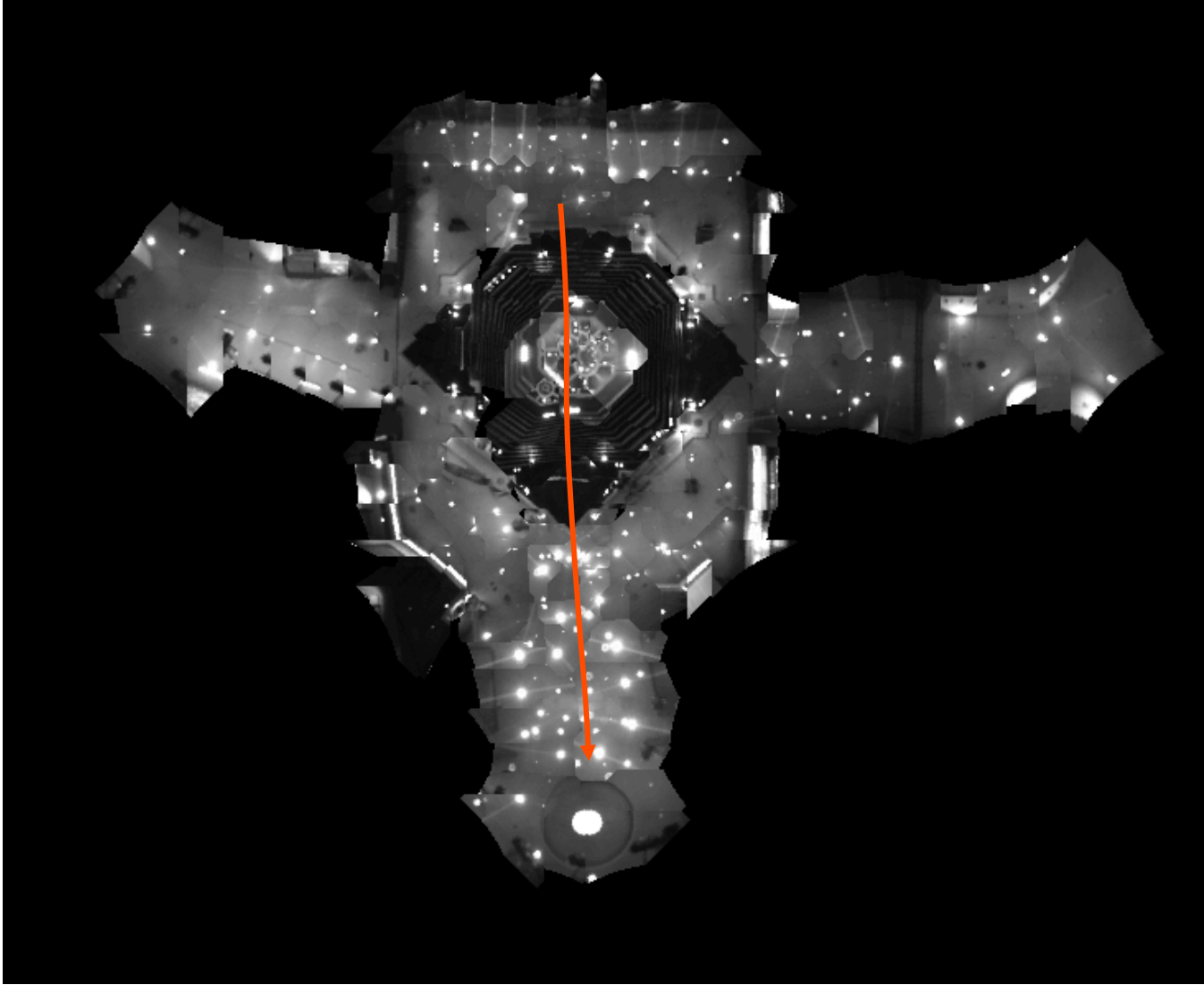
Measurement z :



$P(z|x)$:



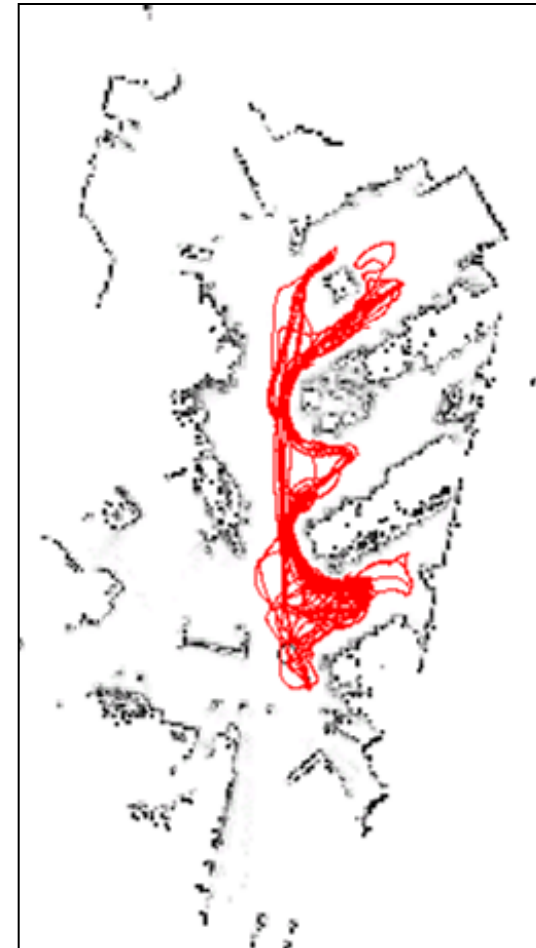
Global Localization Using Vision



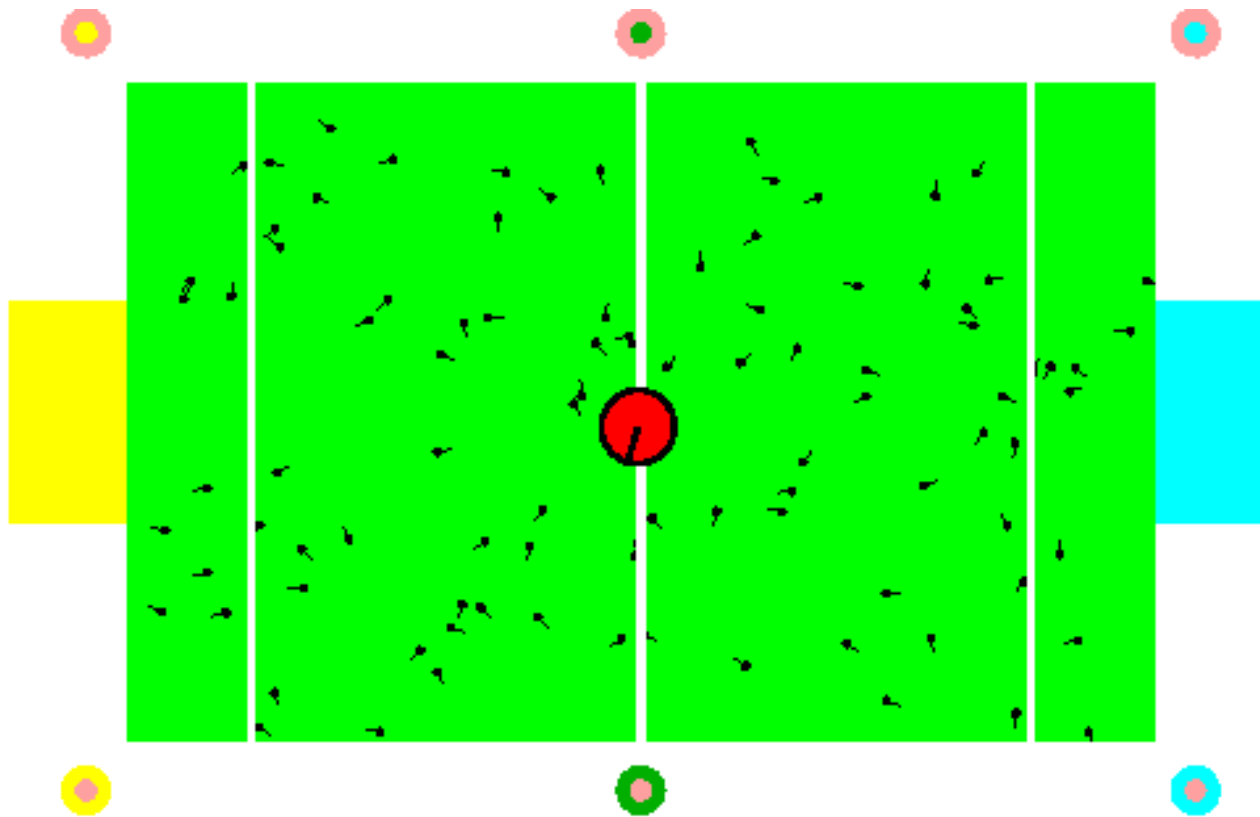
Robots in Action: Albert



Application: Rhino and Albert Synchronized in Munich and Bonn



Localization for AIBO robots



Limitations

- The approach described so far is able to
 - track the pose of a mobile robot and to
 - globally localize the robot.
- How can we deal with localization errors (i.e., the kidnapped robot problem)?

Approaches

- Randomly insert samples (the robot can be teleported at any point in time).
- Insert random samples proportional to the average likelihood of the particles (the robot has been teleported with higher probability when the likelihood of its observations drops).

Random Samples Vision-Based Localization

936 Images, 4MB, .6secs/image

Trajectory of the robot:



Odometry Information



Image Sequence



Resulting Trajectories

Position tracking:

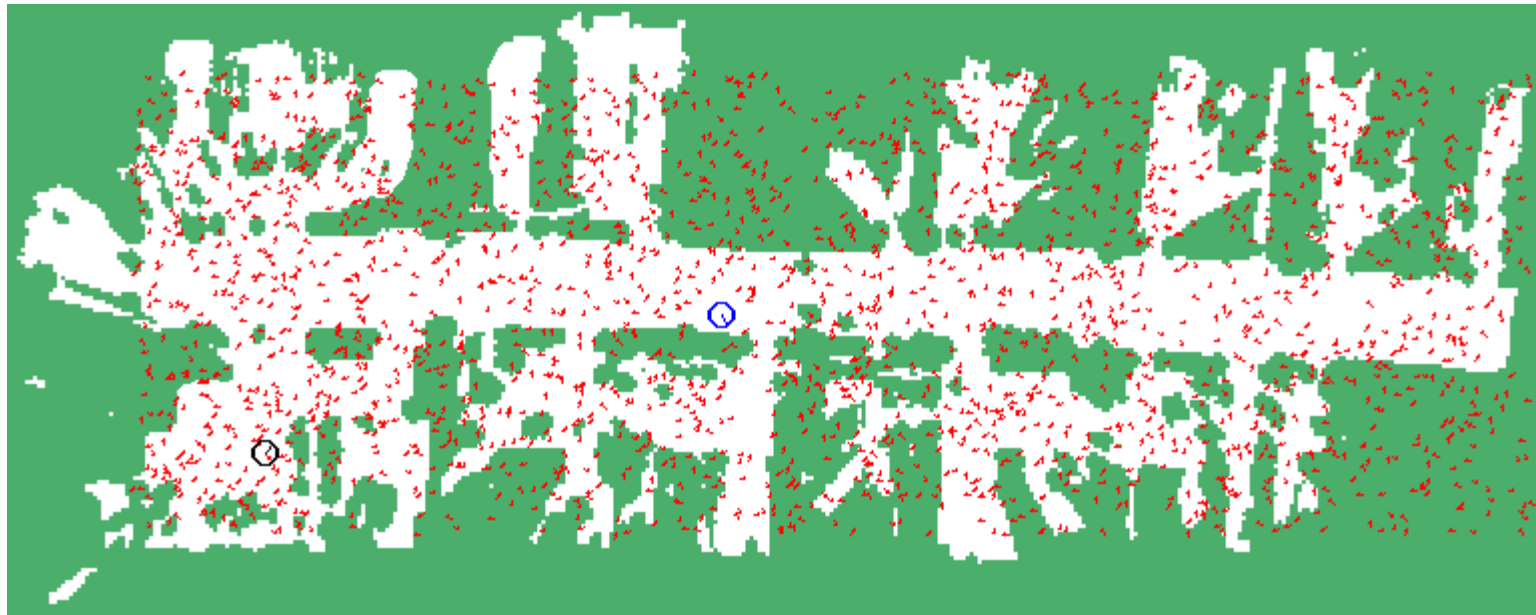


Resulting Trajectories

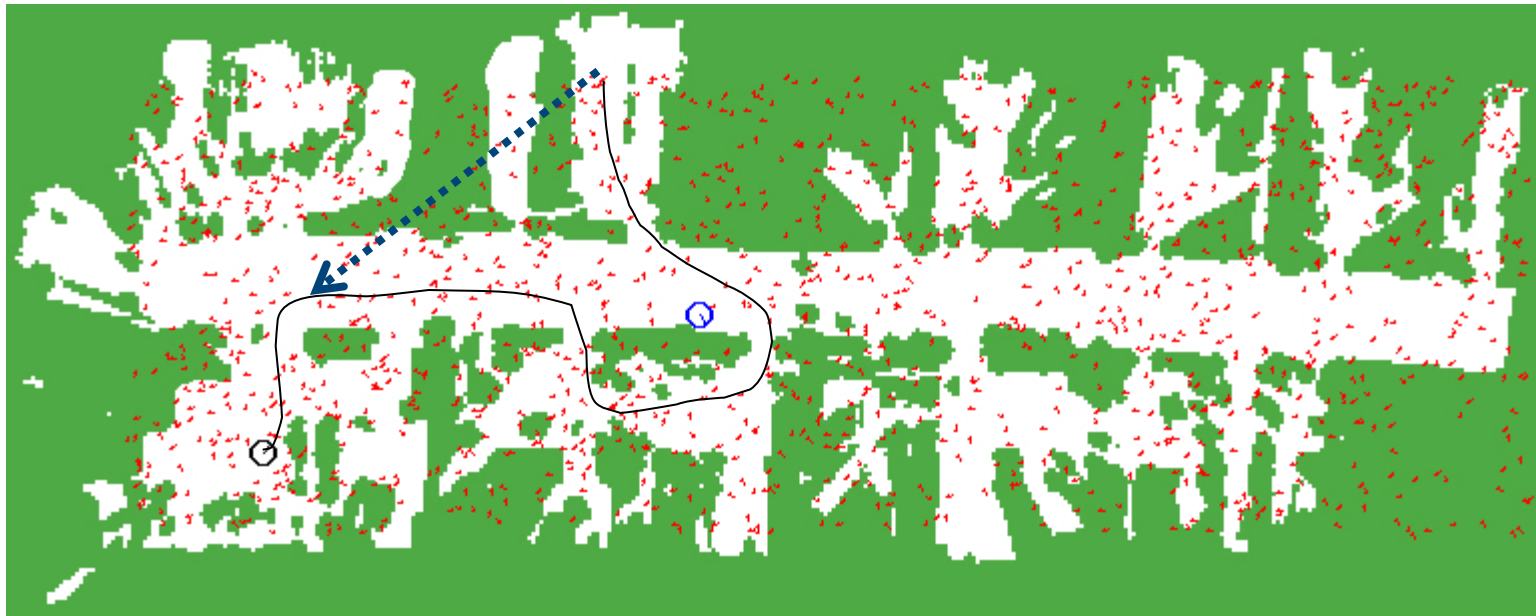
Global localization:



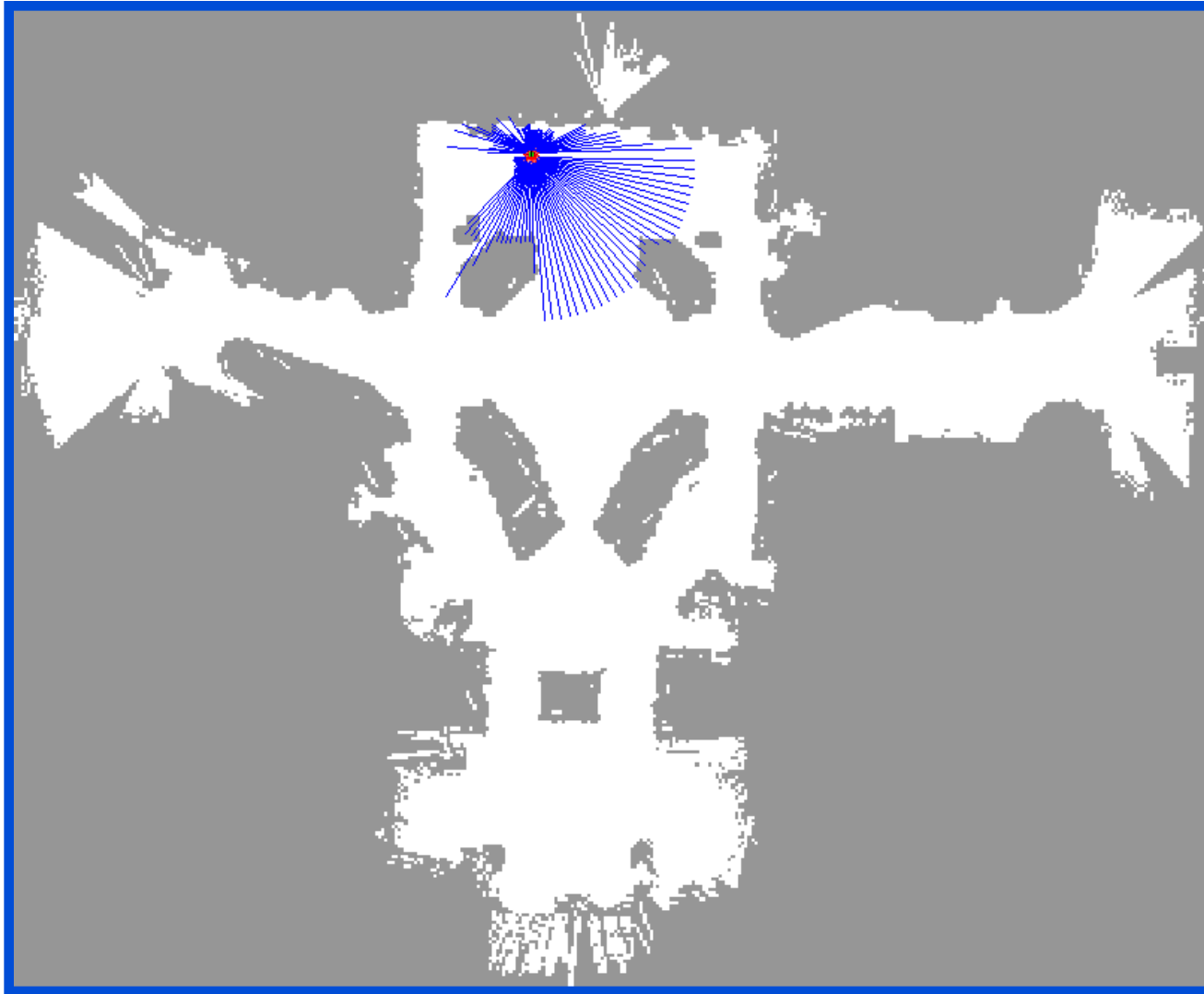
Global Localization



Kidnapping the Robot



Recovery from Failure



Summary

- Particle filters are an implementation of recursive Bayesian filtering
- They represent the posterior by a set of weighted samples.
- In the context of localization, the particles are propagated according to the motion model.
- They are then weighted according to the likelihood of the observations.
- In a re-sampling step, new particles are drawn with a probability proportional to the likelihood of the observation.