CSE-571
Robotics

Fast-SLAM Mapping

Particle Representation

- A set of weighted samples

$$\mathcal{X} \;=\; \left\{\left\langle x^{[i]}, w^{[i]}\right\rangle\right\}_{i=1,\ldots,N}$$

- Think of a sample as one hypothesis about the state
- For feature-based SLAM:

$$x \;=\; (\underbrace{x_{1:t}}_{\text{poses}}, \underbrace{m_{1,x}, m_{1,y}, \ldots, m_{M,x}, m_{M,y}}_{\text{landmarks}})^T$$

Courtesy: C. Stachniss

Dimensionality Problem

Particle filters are effective in low dimensional spaces as the likely regions of the state space need to be covered with samples.

$$x = (\underbrace{x_{1:t}, m_{1,x}, m_{1,y}, \ldots, m_{M,x}, m_{M,y}}_{\text{high-dimensional}})^T$$

Courtesy: C. Stachniss

Can We Exploit Dependencies Between the Different Dimensions of the State Space?

$$x_{1:t}, m_1, \ \ldots, m_M$$

Courtesy: C. Stachniss

If We Know the Poses of the Robot, Mapping is Easy!

$$x_{1:t}, m_1, \ \ldots, m_M$$

Courtesy: C. Stachniss

Key Idea

$$\underline{x_{1:t}}, \underline{m_1, \ \ldots, m_M}$$

If we use the particle set only to model the robot's path, each sample is a path hypothesis. For each sample, we can compute an individual map of landmarks.

Courtesy: C. Stachniss

Rao-Blackwellization

- Factorization to exploit dependencies between variables:

$$p(a, b) = p(b \mid a)\, p(a)$$

- If $p(b \mid a)$ can be computed efficiently, represent only $p(a)$ with samples and compute $p(b \mid a)$ for every sample

Courtesy: C. Stachniss

Rao-Blackwellization for SLAM

- Factorization of the SLAM posterior

poses map observations & movements

$$p(x_{0:t}, m_{1:M} \mid z_{1:t}, u_{1:t}) =$$

First introduced for SLAM by Murphy in 1999

K. Murphy, Bayesian map learning in dynamic environments, In Proc. Advances in Neural Information Processing Systems, 1999

Courtesy: C. Stachniss

Rao-Blackwellization for SLAM

- Factorization of the SLAM posterior

poses map observations & movements

$$p(x_{0:t}, m_{1:M} \mid z_{1:t}, u_{1:t}) =$$
$$p(x_{0:t} \mid z_{1:t}, u_{1:t})\, p(m_{1:M} \mid x_{0:t}, z_{1:t})$$

path posterior map posterior

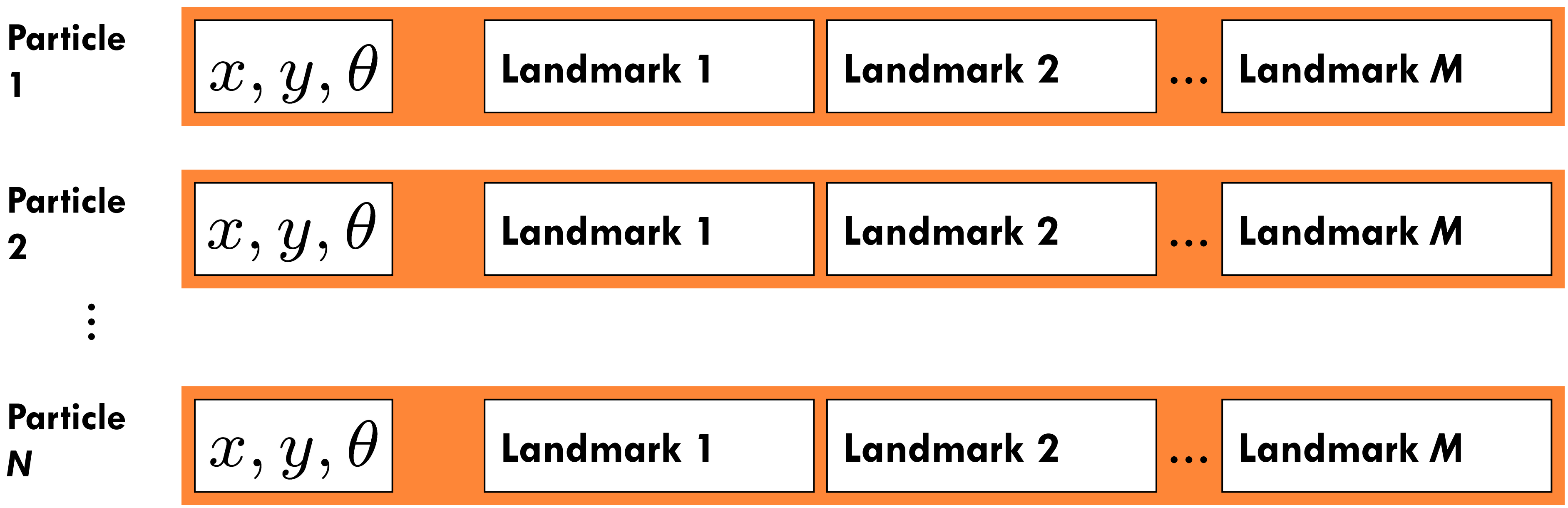
First introduced for SLAM by Murphy in 1999

K. Murphy, Bayesian map learning in dynamic environments, In Proc. Advances in Neural Information Processing Systems, 1999

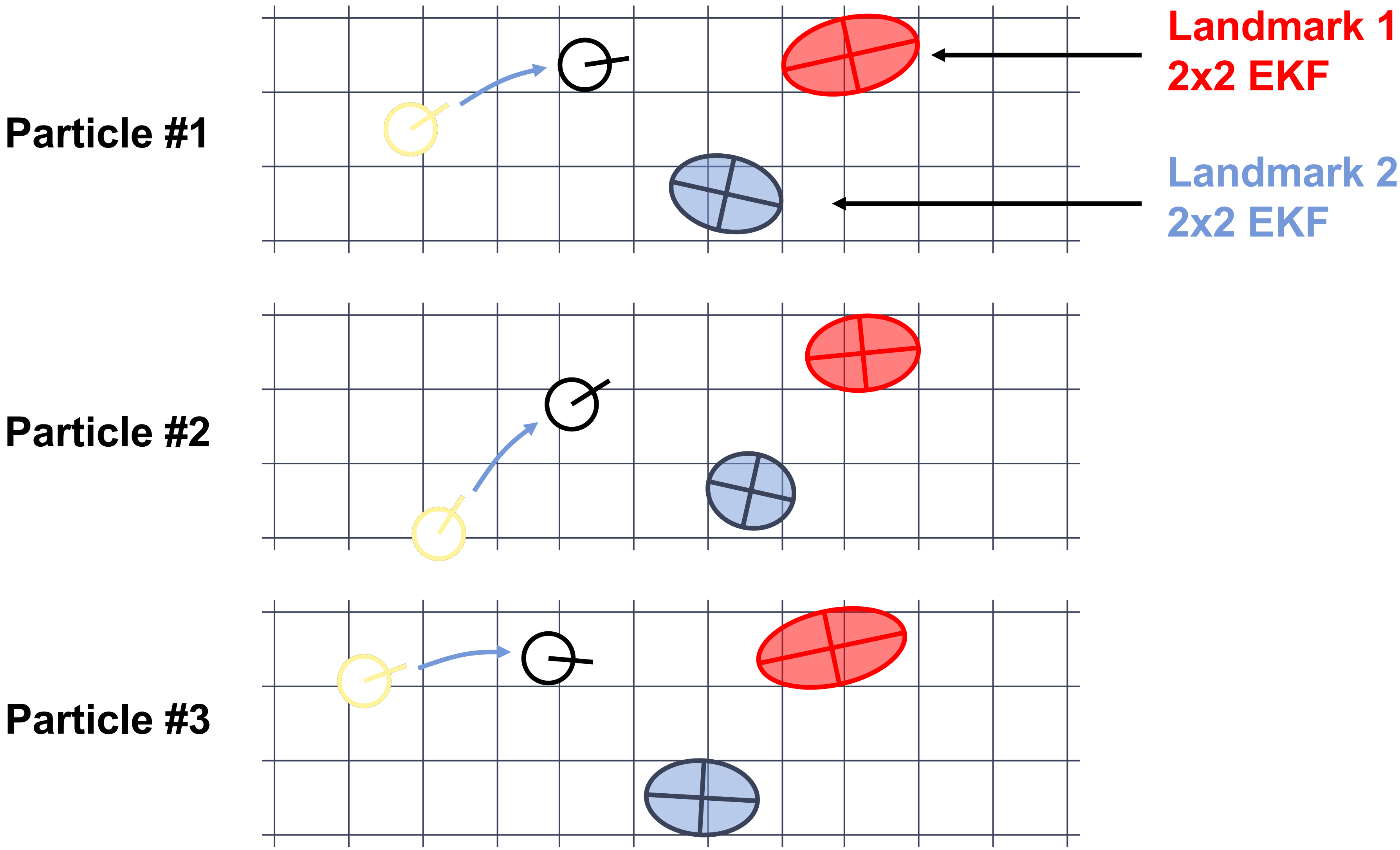
Courtesy: C. Stachniss

FastSLAM

- Proposed by Montemerlo et al. in 2002
- Each landmark is represented by a 2x2 EKF
- Each particle therefore has to maintain *M* individual EKFs

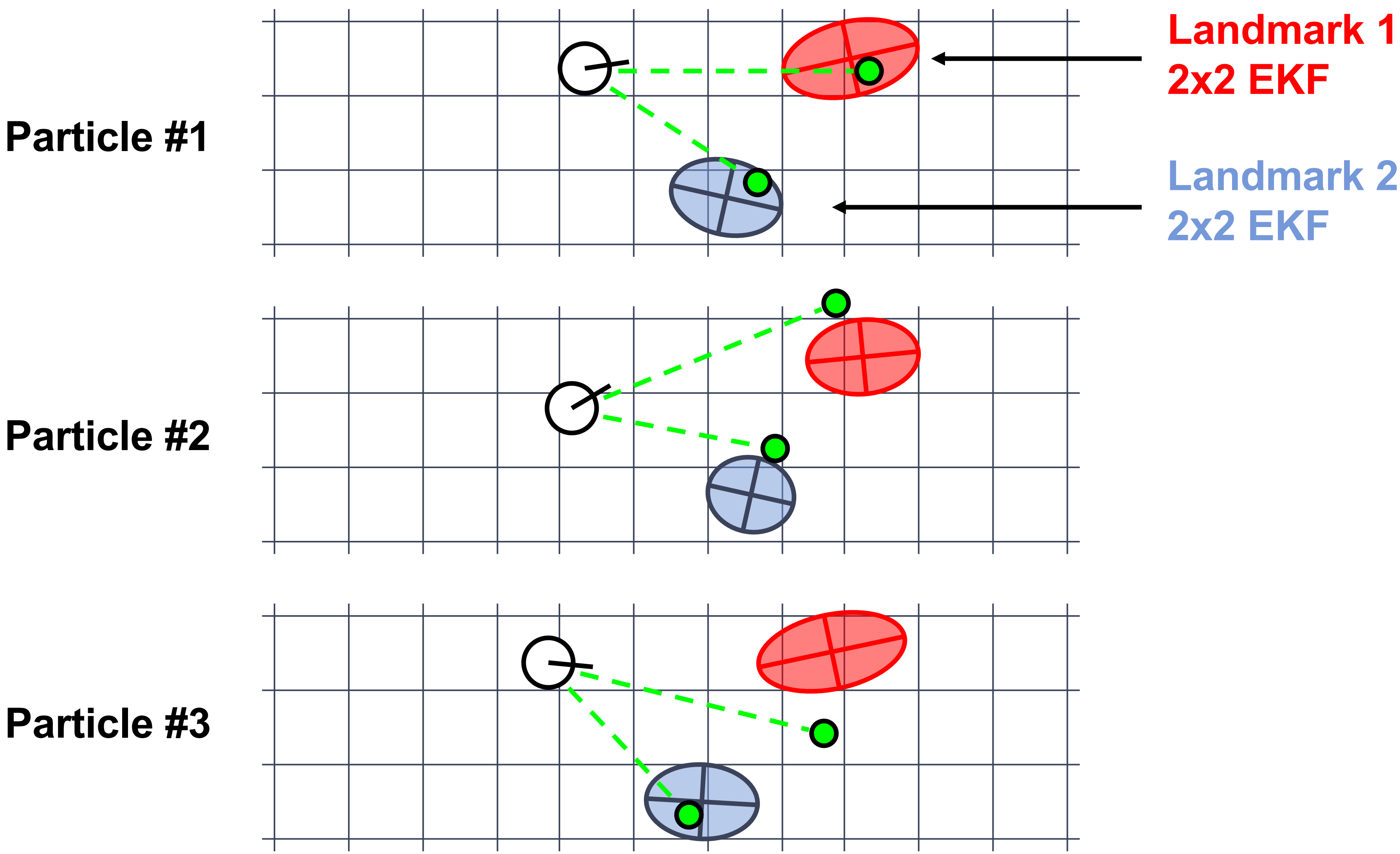


FastSLAM – Motion Update



Courtesy: M. Montemerlo

FastSLAM – Sensor Update



Courtesy: M. Montemerlo

FastSLAM – Sensor Update

Particle #1

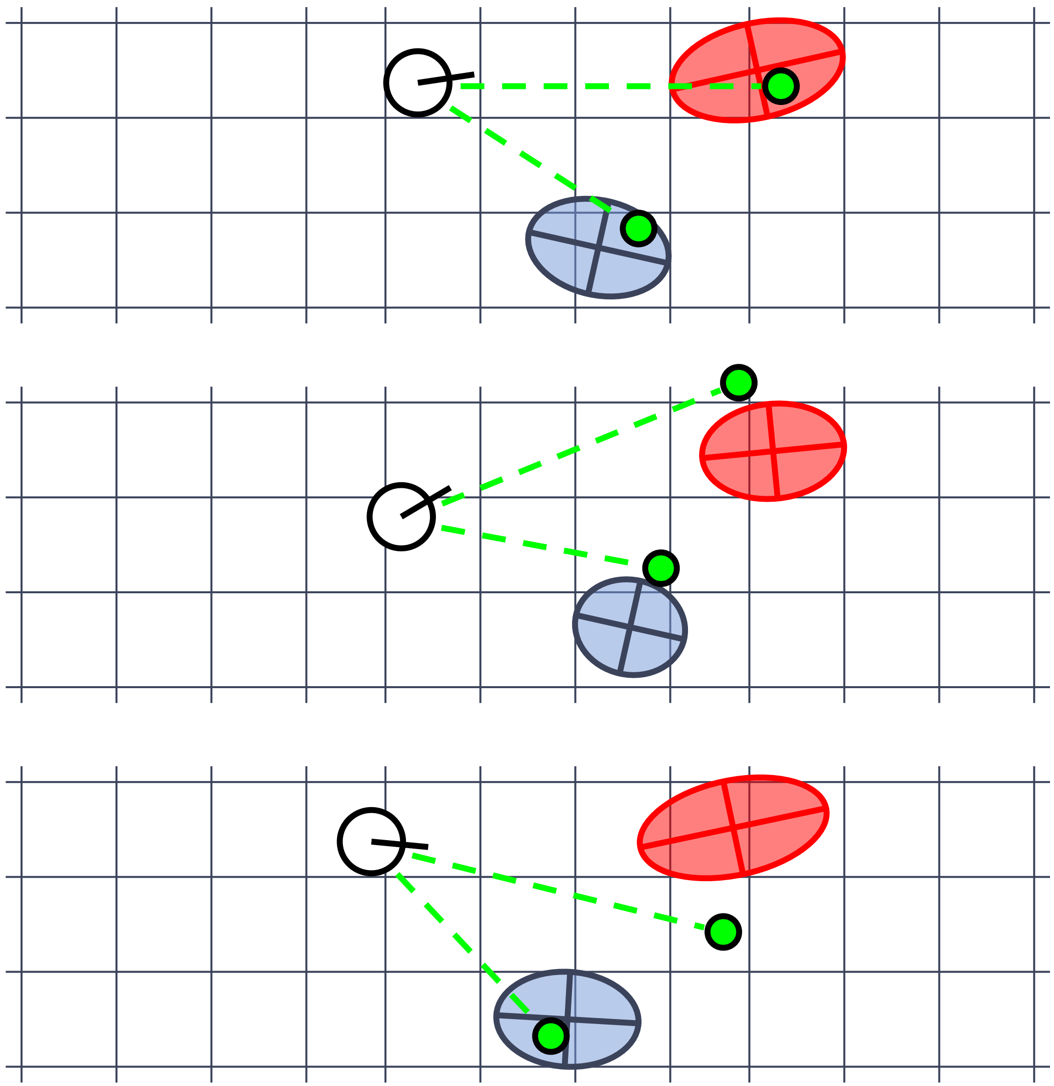
Weight = 0.8

Particle #2

Weight = 0.4

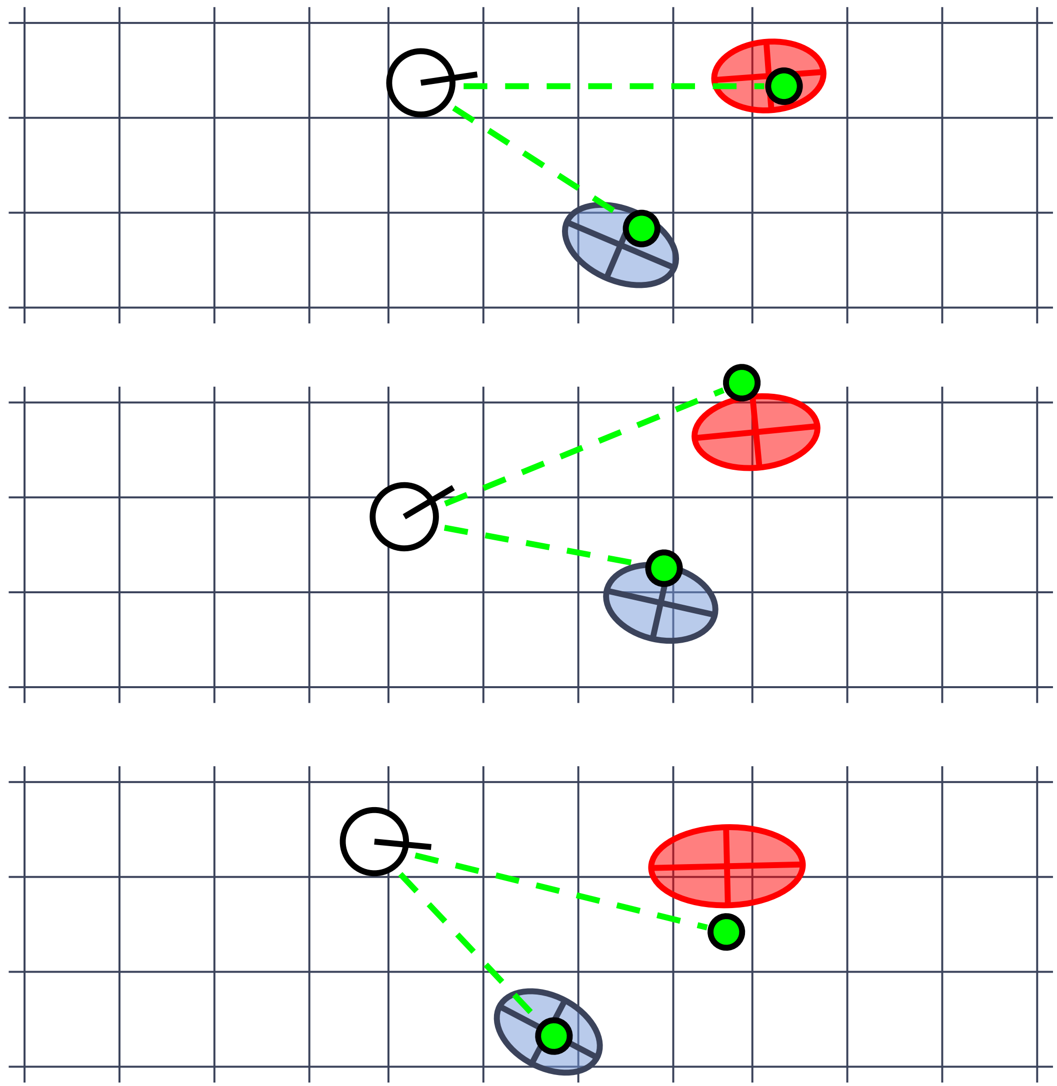
Particle #3

Weight = 0.1



Courtesy: M. Montemerlo

FastSLAM – Sensor Update



Particle #1 — **Update map of particle 1**

Particle #2 — **Update map of particle 2**

Particle #3 — **Update map of particle 3**

Courtesy: M. Montemerlo

Key Steps of FastSLAM 1.0

- Extend the path posterior by sampling a new pose for each sample

$$x_t^{[k]} \sim p(x_t \mid x_{t-1}^{[k]}, u_t)$$

- Compute particle weight

exp. observation

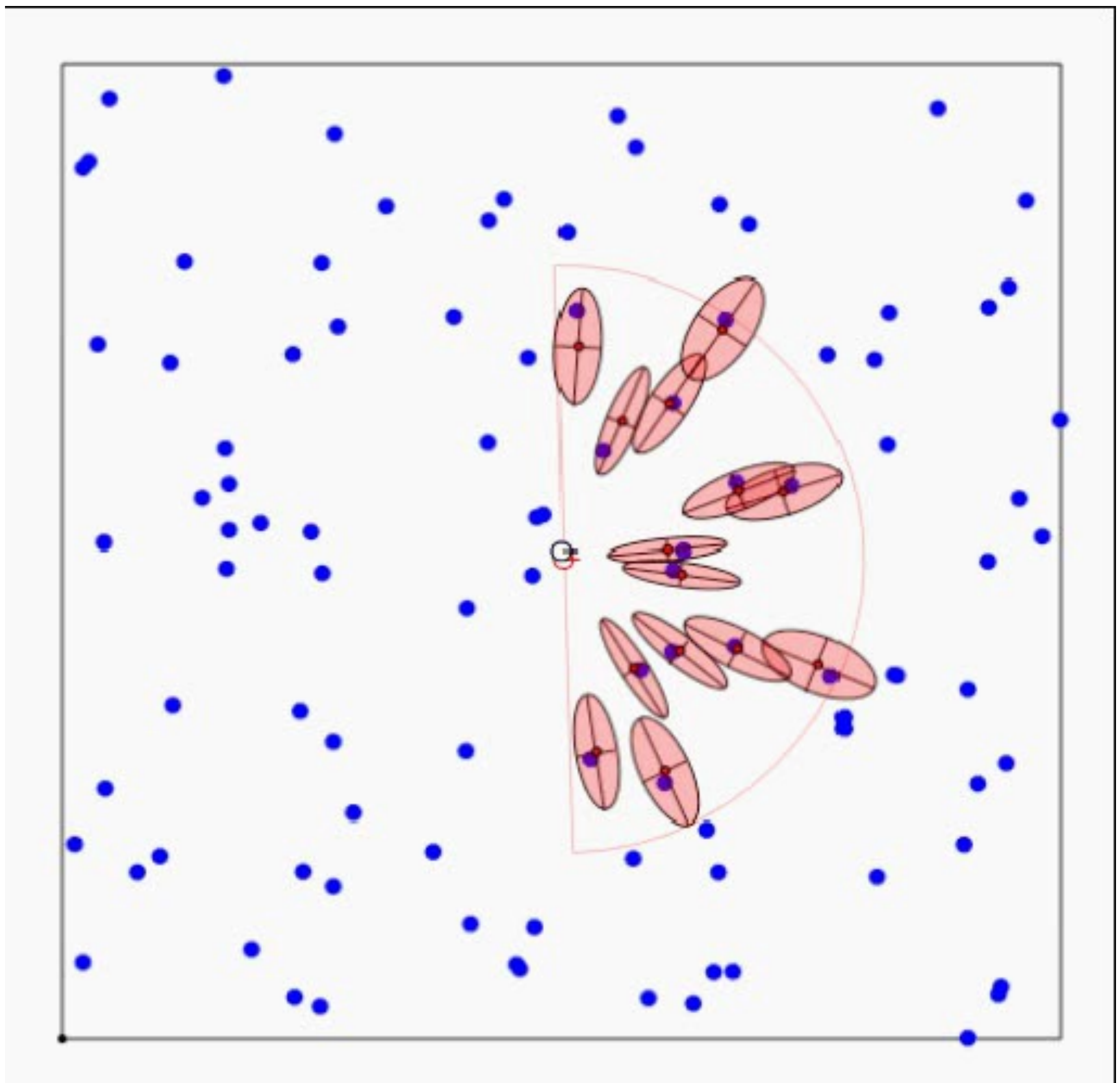
$$w^{[k]} = |2\pi Q|^{-\frac{1}{2}} \ \exp\left\{-\frac{1}{2}(z_t - \hat{z}^{[k]})^T Q^{-1} \ (z_t - \hat{z}^{[k]})\right\}$$

innovation covariance

- Update belief of observed landmarks (EKF update rule)
- Resample

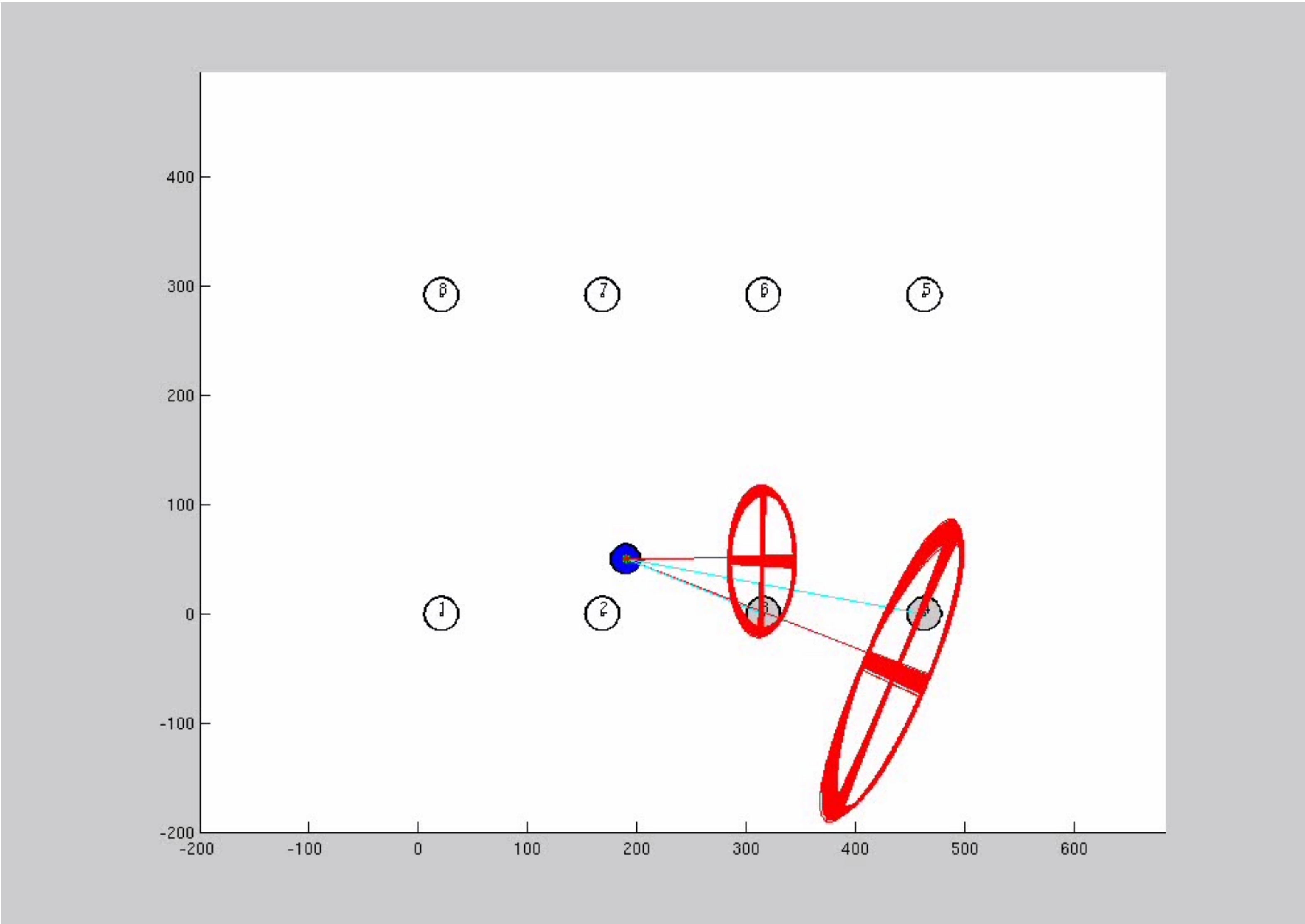
Courtesy: C. Stachniss

FastSLAM in Action

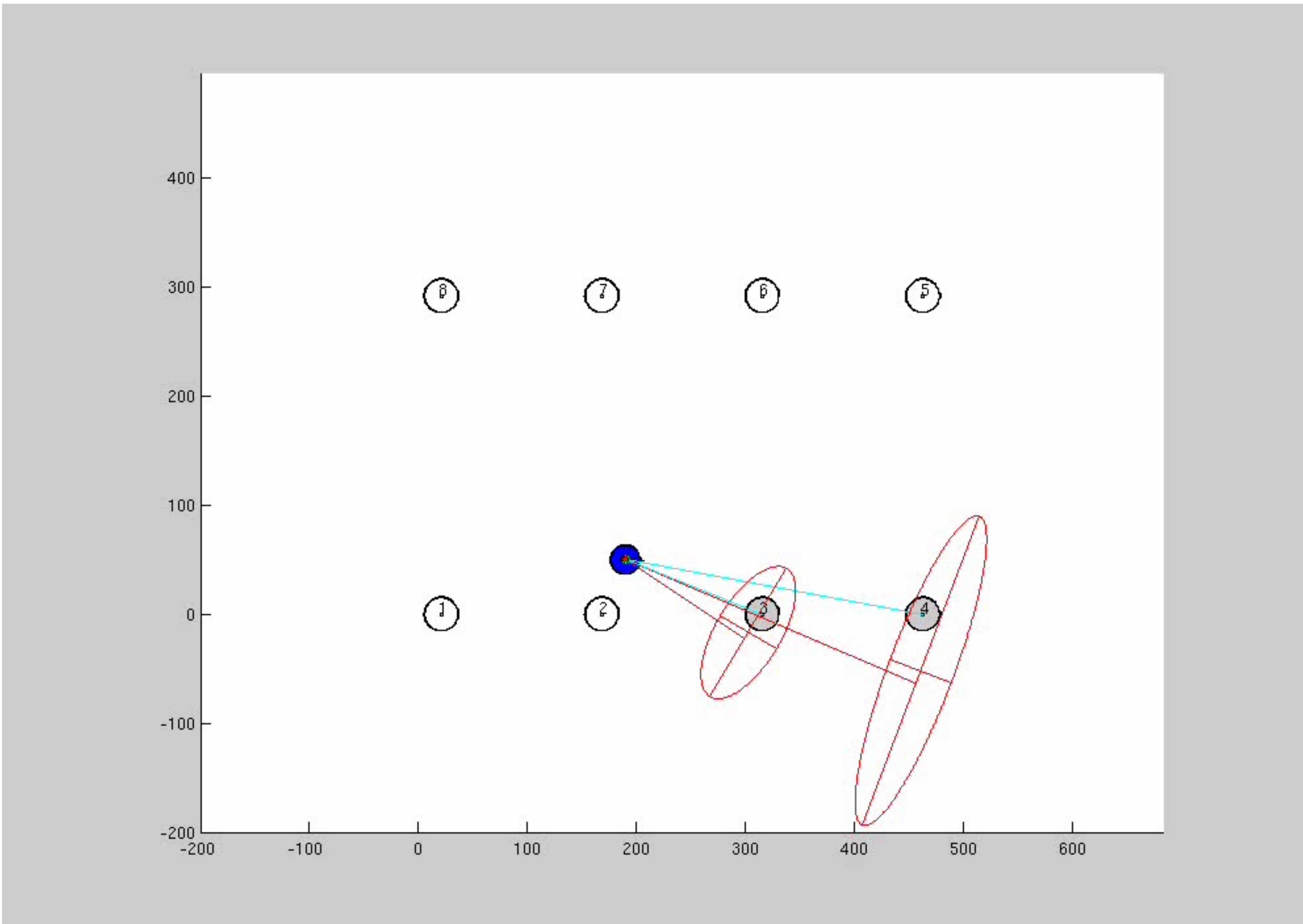


Courtesy: M. Montemerlo

FastSLAM – Video – All Maps

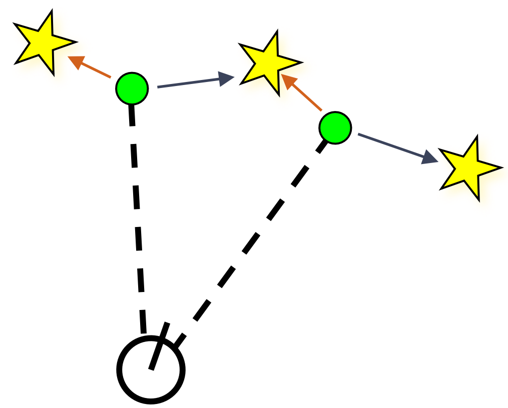


FastSLAM – Video – “Best” particle in terms of Cum Log Prob



Data Association Problem

- Which observation belongs to which landmark?

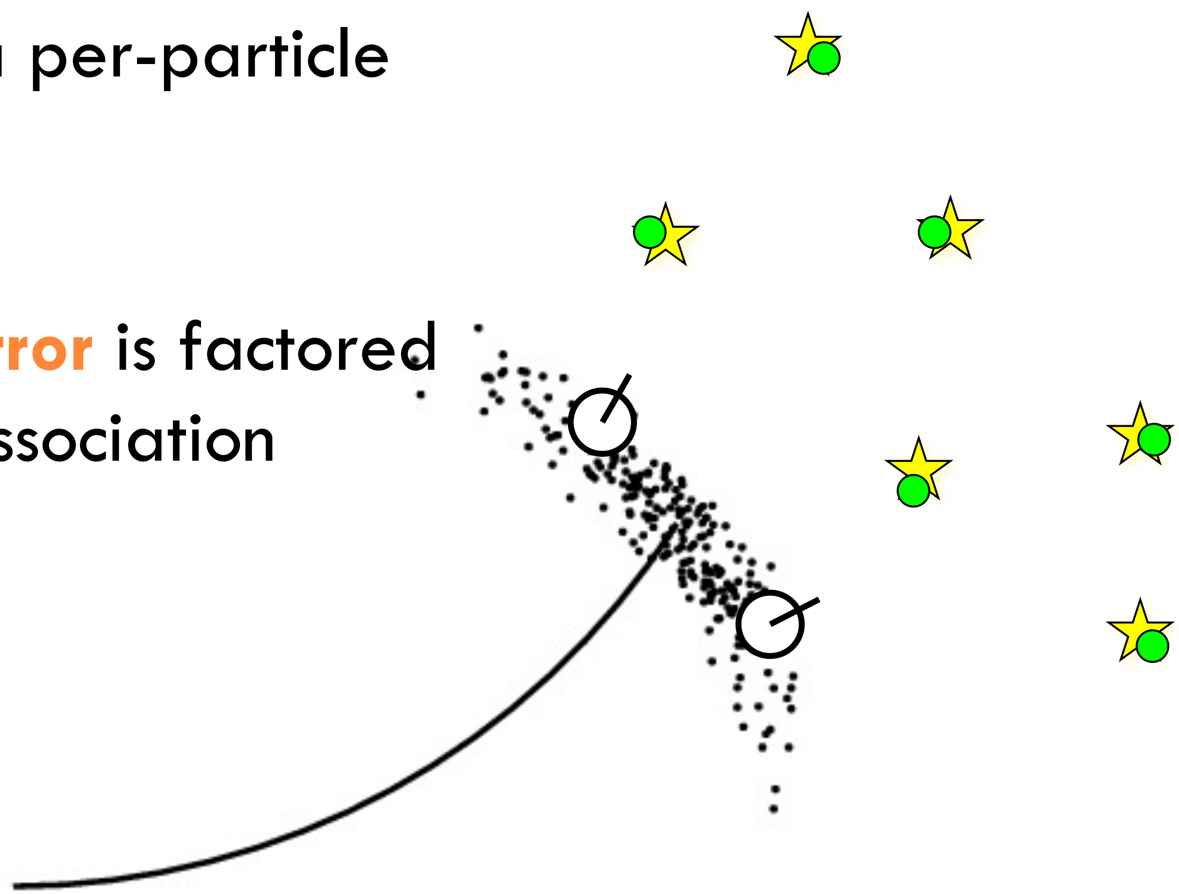


- More than one possible association
- **Potential data associations depend on the pose of the robot**

Courtesy: M. Montemerlo

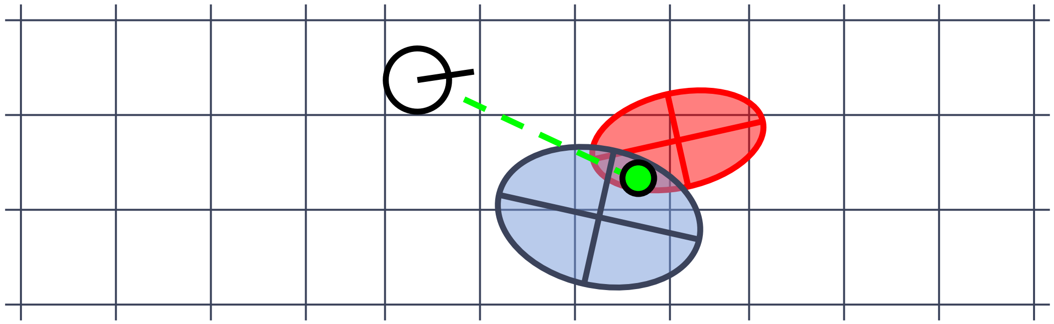
Particles Support for Multi-Hypotheses Data Association

- Decisions on a per-particle basis
- Robot pose **error** is factored out of data association decisions



Courtesy: M. Montemerlo

Per-Particle Data Association

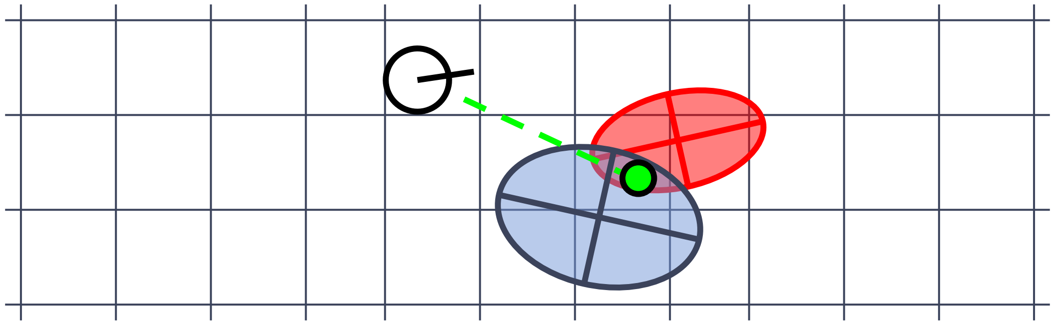


Was the observation generated by the **red** or by the **blue** landmark?

P(observation | red) = 0.3 P(observation | blue) = 0.7

Courtesy: M. Montemerlo

Per-Particle Data Association



Was the observation generated by the **red** or by the **blue** landmark?

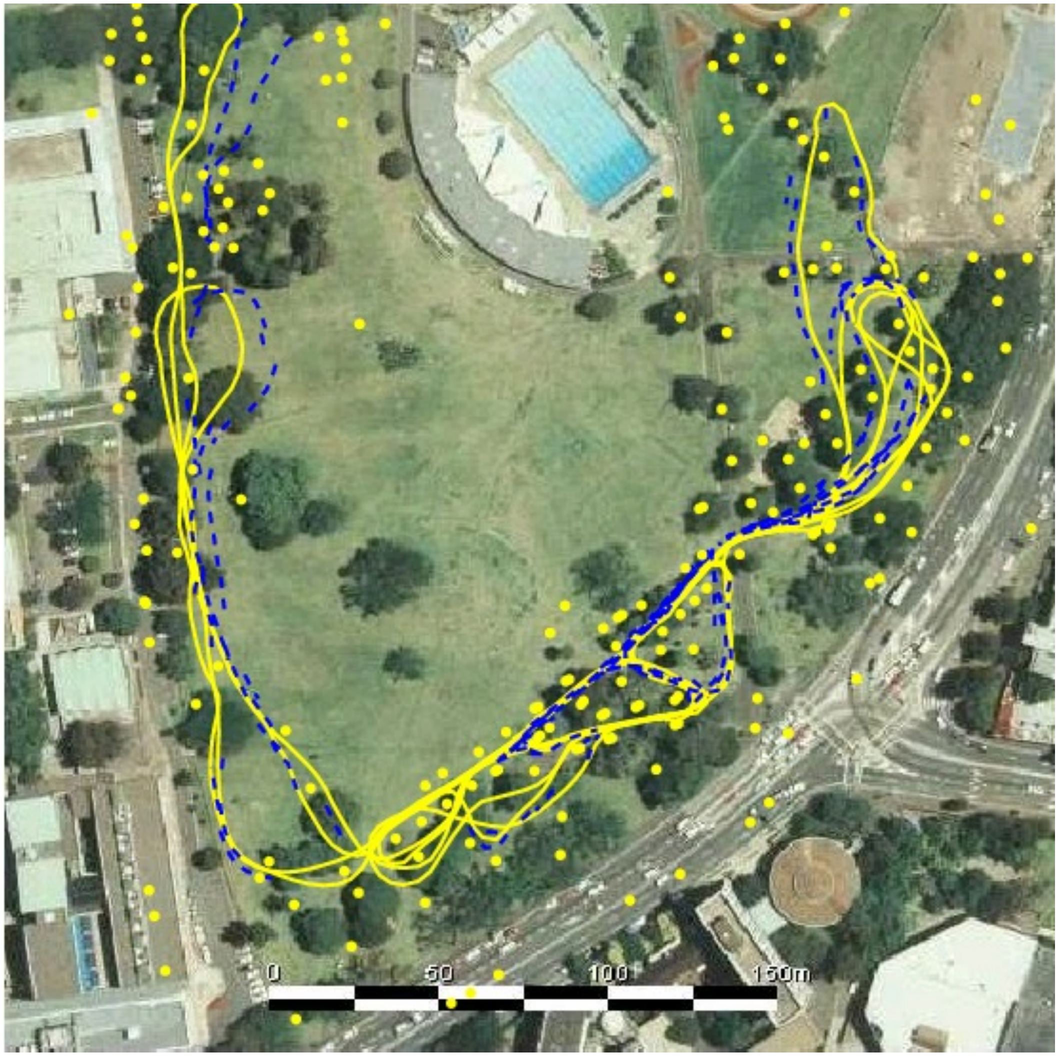
P(observation | red) = 0.3   P(observation | blue) = 0.7

- Two options for per-particle data association
  - Pick the most probable match
  - Pick a random association weighted by the observation likelihoods
- If the probability for an assignment is too low, generate a new landmark

Courtesy: M. Montemerlo

Results – Victoria Park

- 4 km traverse
- < 2.5 m RMS position error
- 100 particles



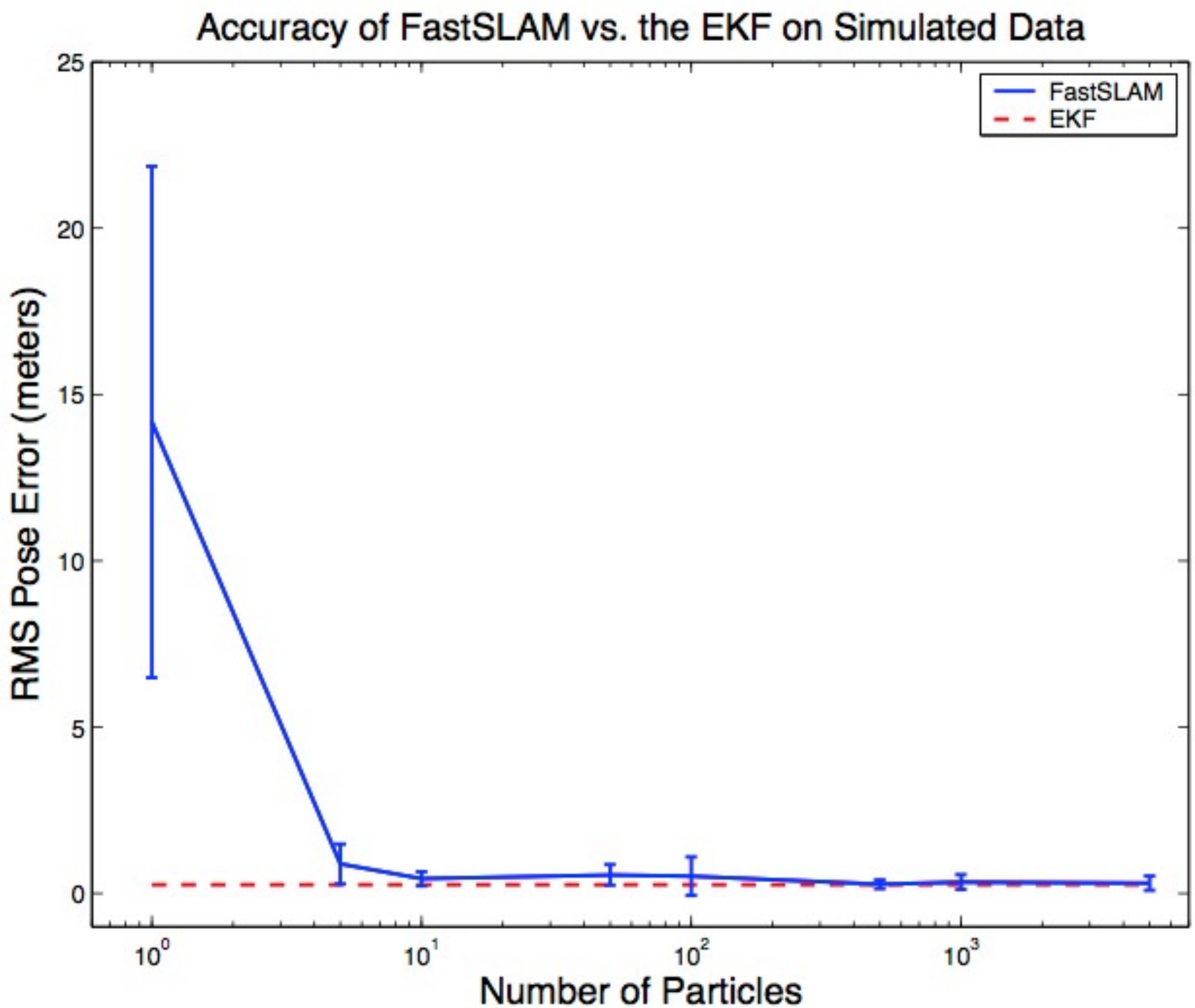
Blue = GPS
Yellow = FastSLAM

Courtesy: M. Montemerlo

Results – Victoria Park (Video)

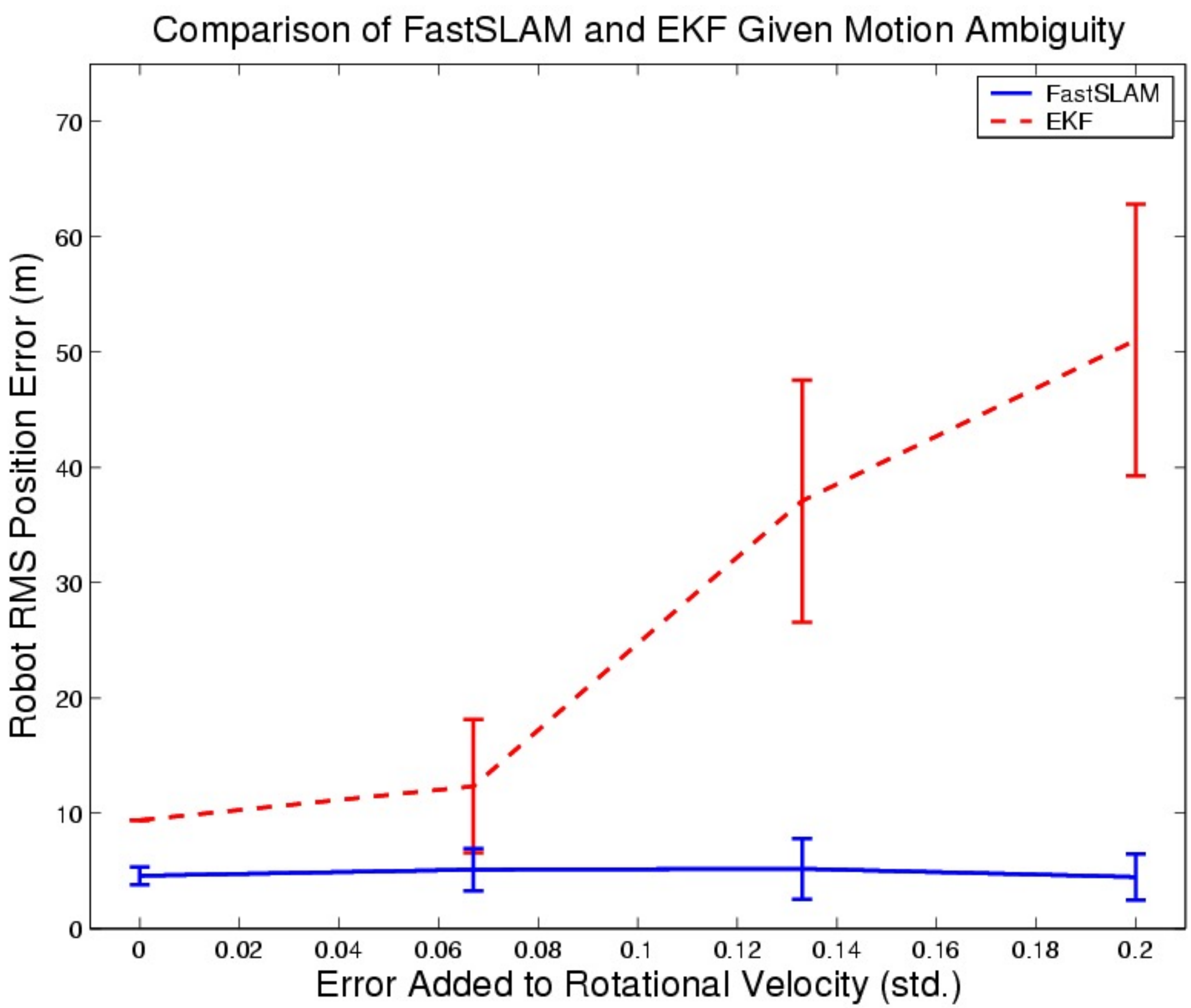
Courtesy: M. Montemerlo

Results (Sample Size)



Courtesy: M. Montemerlo

Results (Motion Uncertainty)



Courtesy: M. Montemerlo

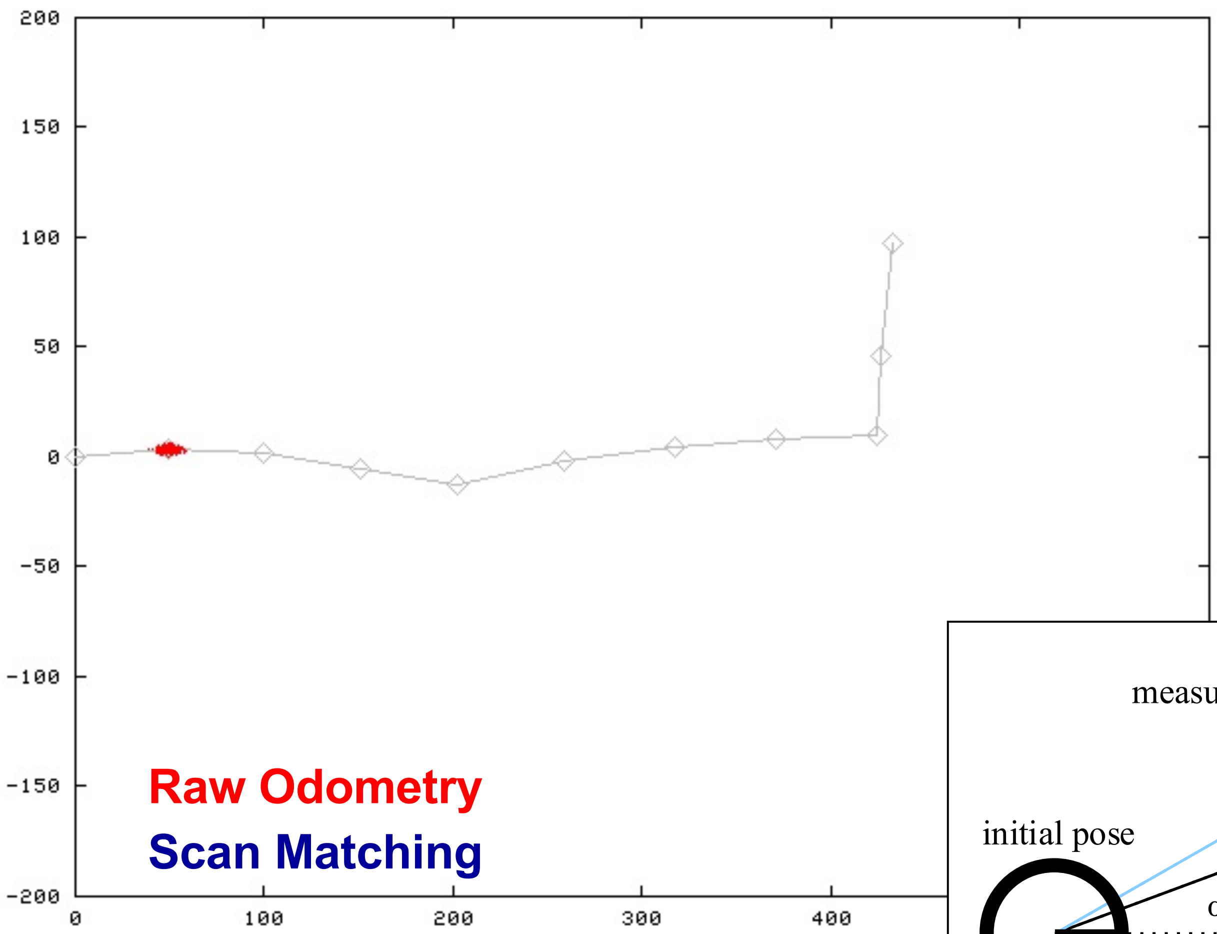
Techniques to Reduce the Number of Particles Needed

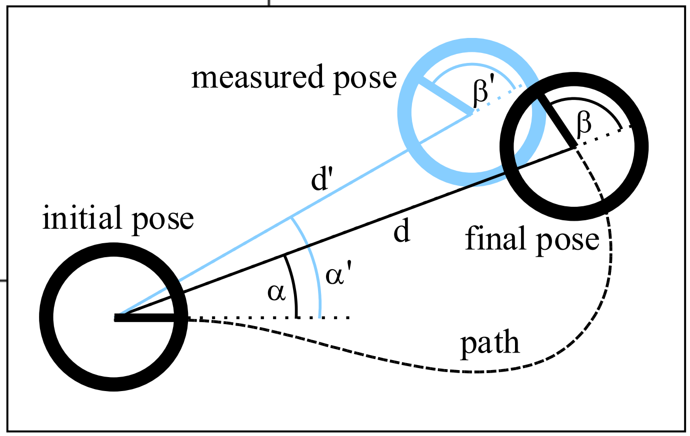
- Better proposals (put the particles in the right place in the prediction step).
- Avoid particle depletion (re-sample only when needed).

Generating better Proposals

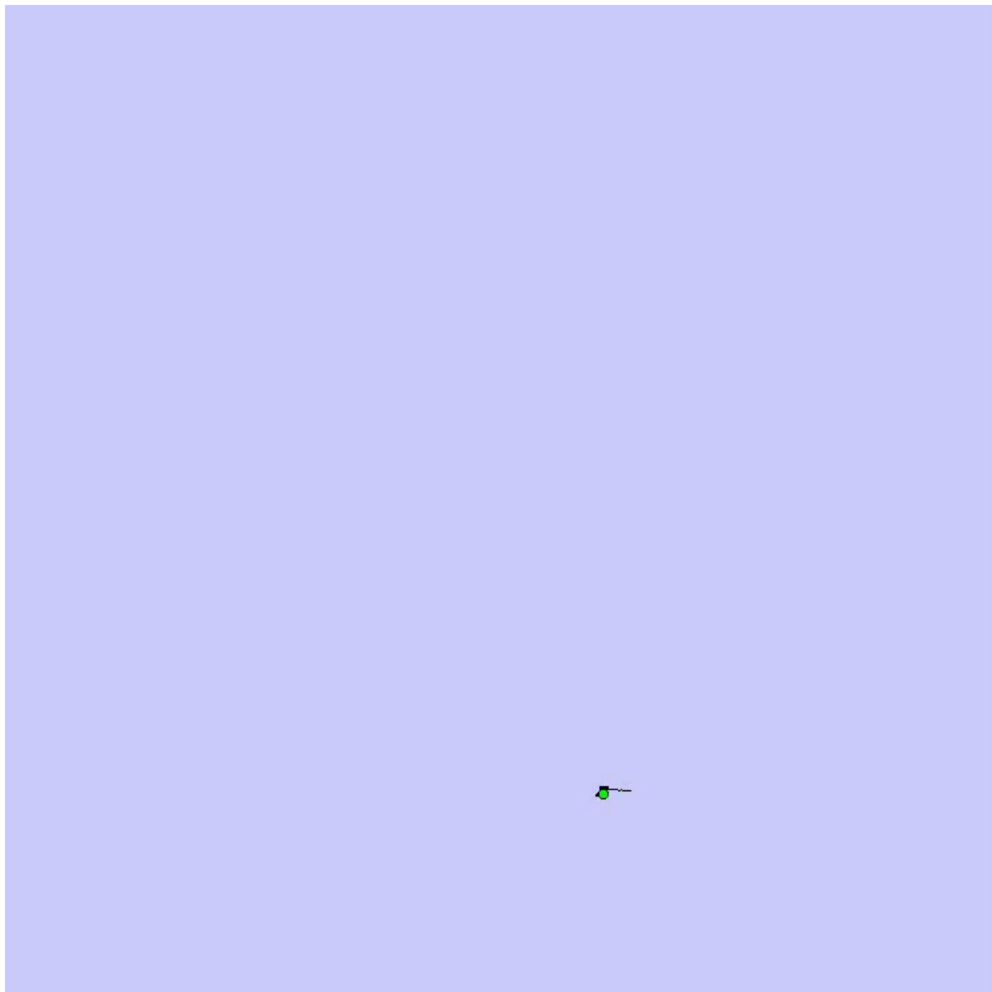
- Use scan-matching to compute highly accurate odometry measurements from consecutive range scans.
- Use the improved odometry in the prediction step to get highly accurate proposal distributions.

Motion Model for Scan Matching



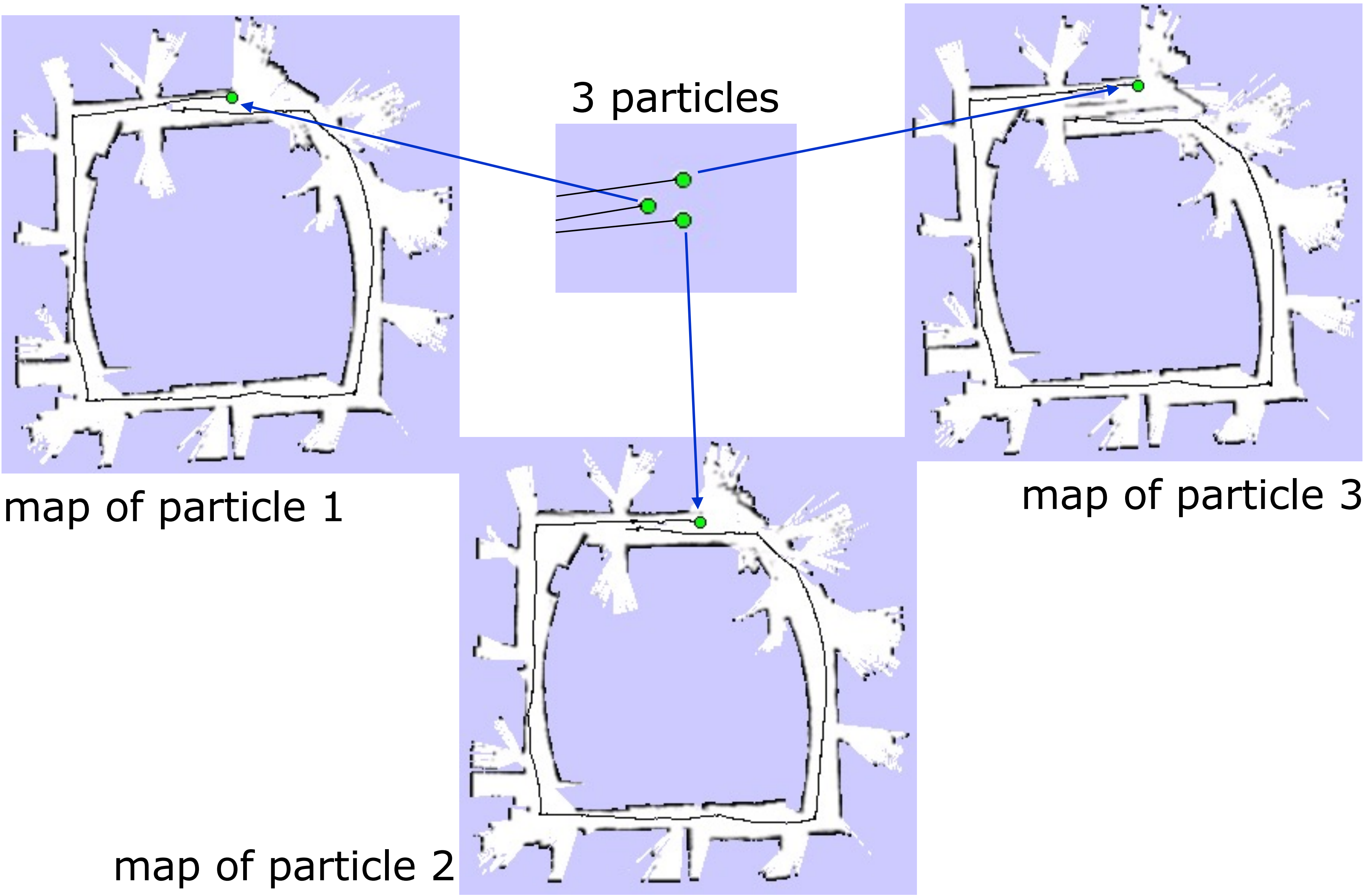


Rao-Blackwellized Mapping with Scan-Matching

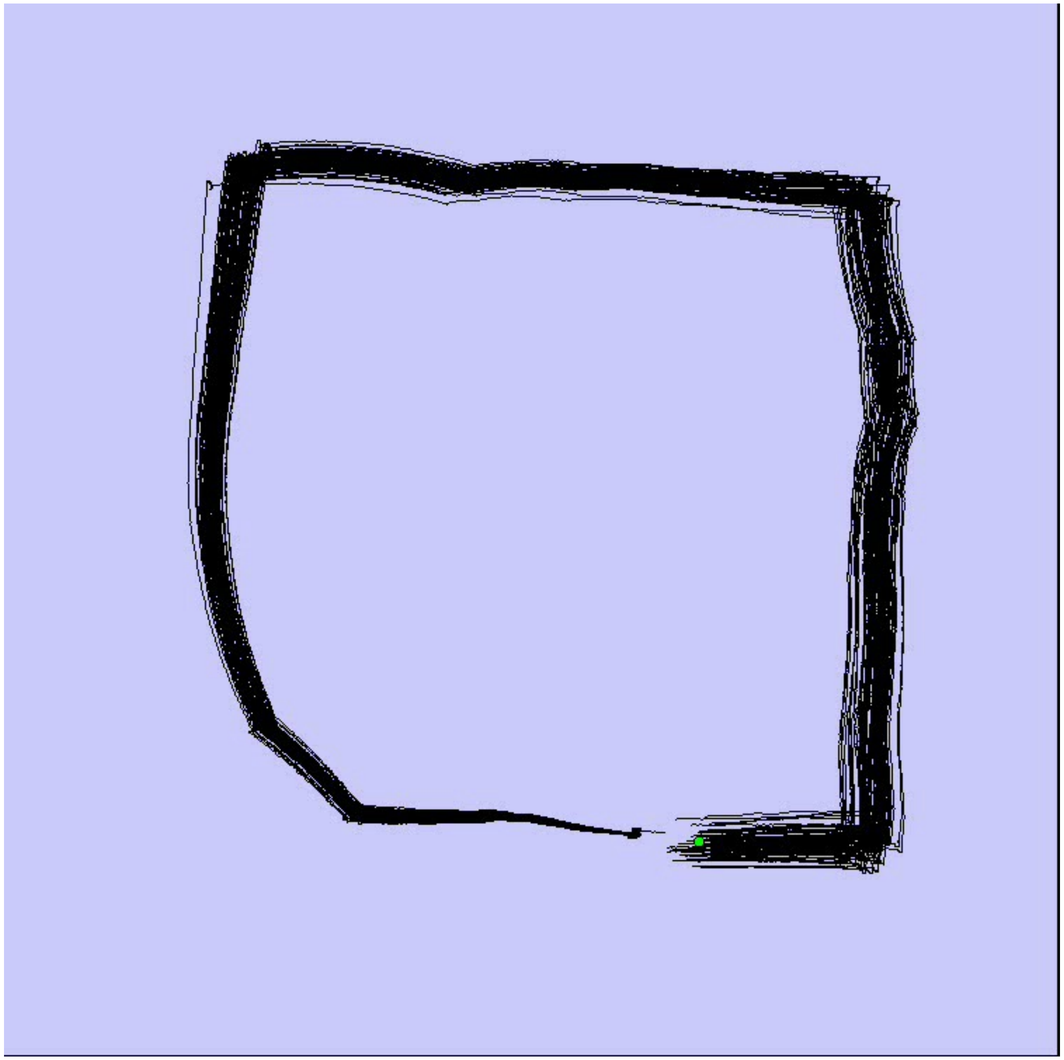


Map: Intel Research Lab Seattle

Loop Closure Example

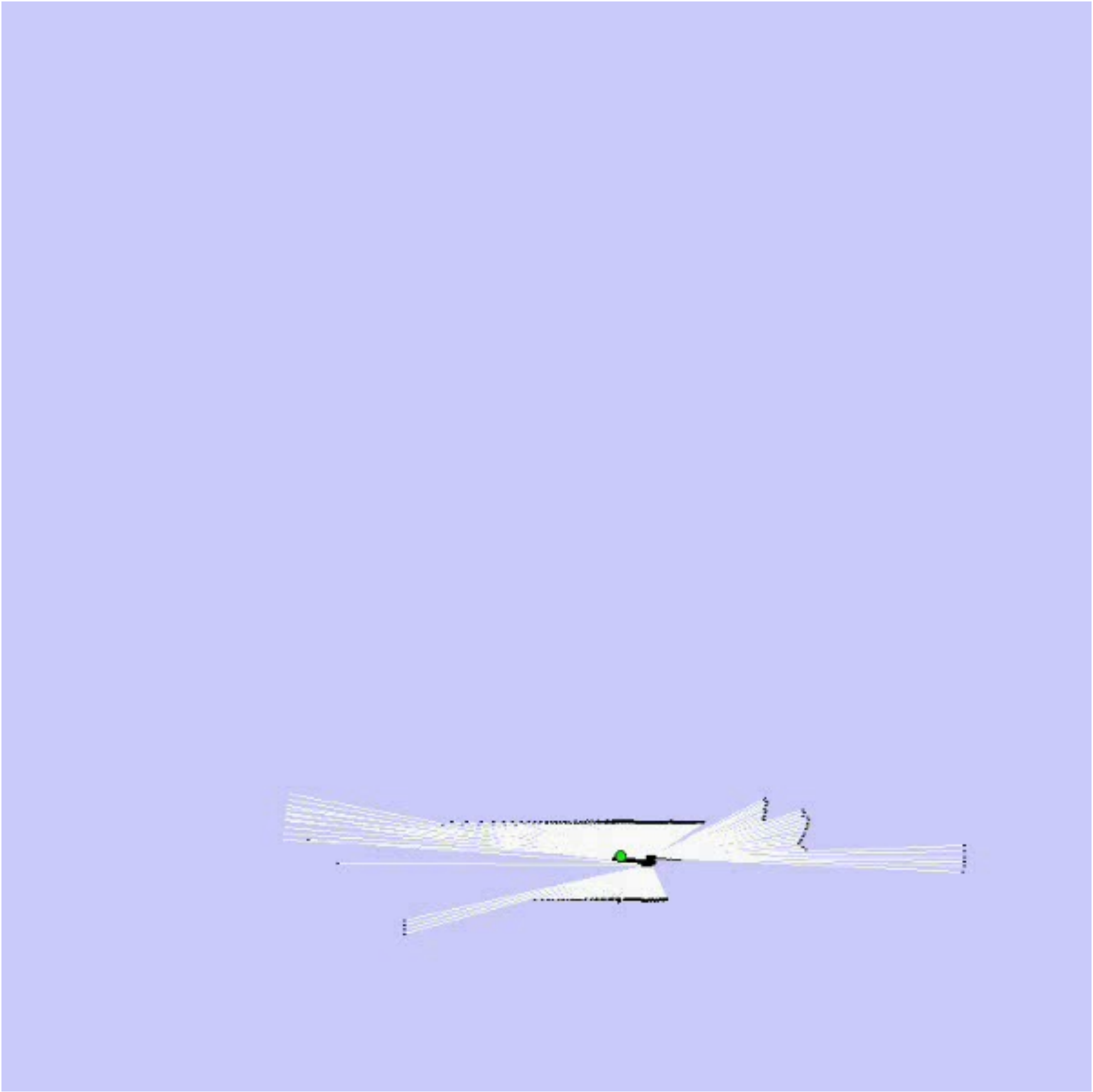


Rao-Blackwellized Mapping with Scan-Matching



Map: Intel Research Lab Seattle

Rao-Blackwellized Mapping with Scan-Matching



Map: Intel Research Lab Seattle

Example (Intel Lab)

- **15 particles**
- four times faster than real-time P4, 2.8GHz
- 5cm resolution during scan matching
- 1cm resolution in final map

Work by Grisetti et al.

Outdoor Campus Map

- **30 particles**
- 250x250m^2
- 1.088 miles (odometry)
- 20cm resolution during scan matching
- 30cm resolution in final map

Work by Grisetti et al.

FastSLAM Summary

- Particle filter-based SLAM
- Rao-Blackwellization: model the robot's path by sampling and compute the landmarks given the poses
- Allow for per-particle data association
- Complexity $\mathcal{O}(N \log M)$

Courtesy: C. Stachniss

Literature

FastSLAM

- Thrun et al.: “Probabilistic Robotics”, Chapter 13.1-13.3 + 13.8 (see errata!)
- Montemerlo, Thrun, Kollar, Wegbreit: FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem, 2002
- Montemerlo and Thrun: Simultaneous Localization and Mapping with Unknown Data Association Using FastSLAM, 2003

Courtesy: C. Stachniss

6D OBJECT POSE ESTIMATION

6D Object Pose

3D Translation

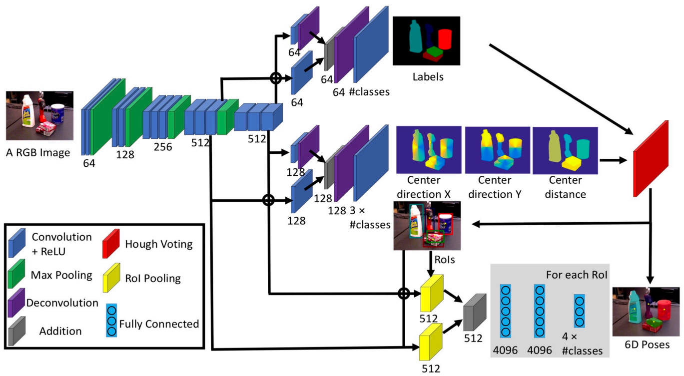
3D Orientation

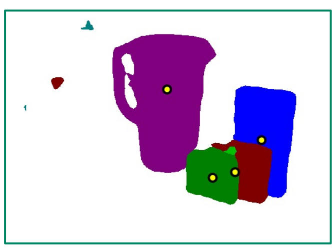
NVIDIA

[Xiang-Schmidt-Narayanan-Fox: RSS-18]

POSE-CNN

Handles symmetric, texture-less objects under partial occlusions

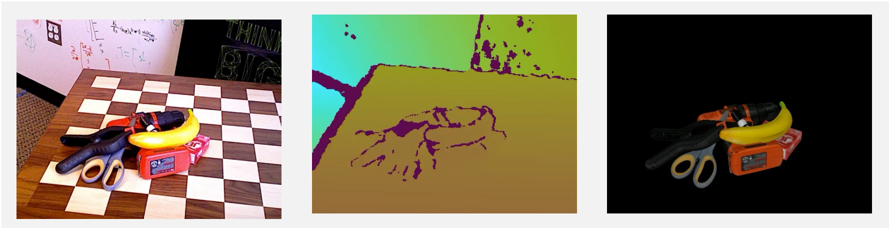




Centers and masks

6D poses

- Provides object mask and 3D position and orientation of object relative to camera
- Operates at 10Hz, sufficient to initialize a tracker
- With ICP, state of the art results on LineMod and YCB-Video



RELATED WORK

Single image

- Object symmetries ignored or special cases
  - Tremblay et al. CoRL 2018 (DOPE)
  - Tekin et al. CVPR 2018
  - Xiang et al. RSS 2018 (PoseCNN)
  - Li et al. ECCV 2018 (DeepIM)
  - Manhardt et al. ECCV 2018

Techniques aim at a unique pose estimate

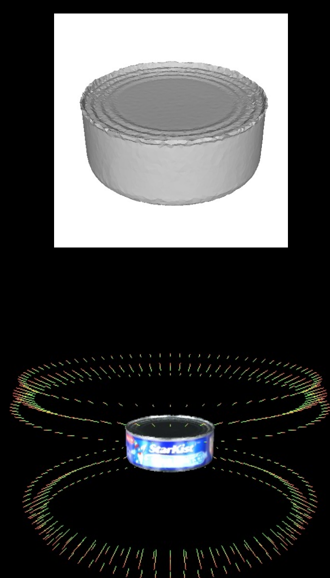
6D object pose tracking

- Unimodal tracking
  - Kehl et al. CVPR 2018
  - Tjaden et al. ICCV 2017
  - Prisacariu et al. IJCV 2017 (PWP3D)
  - Srivatsan et al. RSS 2017
- 6D particle filter
  - Choi et al. IROS 2013

Not designed to estimate multi-modal distributions

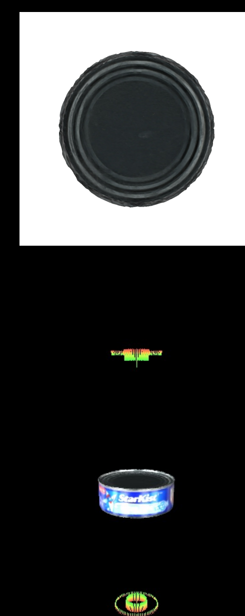
ORIENTATION UNCERTAINTY

Depends on context, shape, sensor



Shape symmetry

Texture breaks symmetry

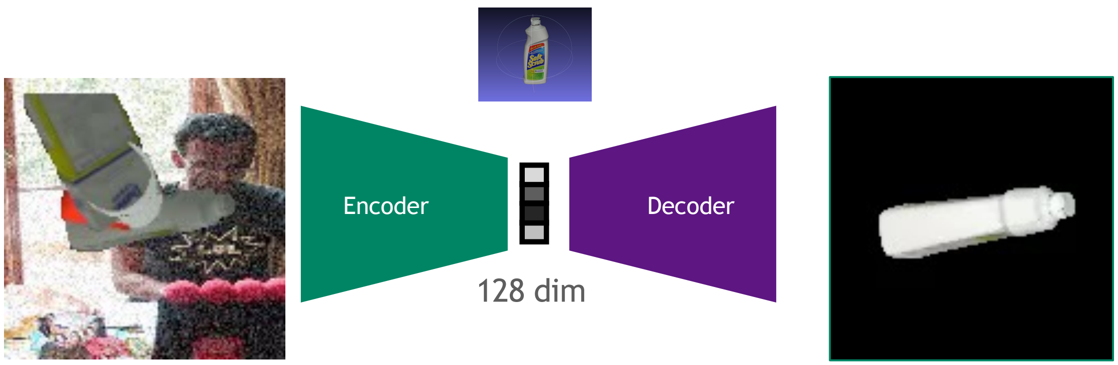


View-based uncertainty

[Sundermeyer-Marton-Durner-Brucker-Triebel: ECCV-18]

TRAINING VIEW-BASED SIMILARITY

Randomly Sample Views onto the Textured Object Model



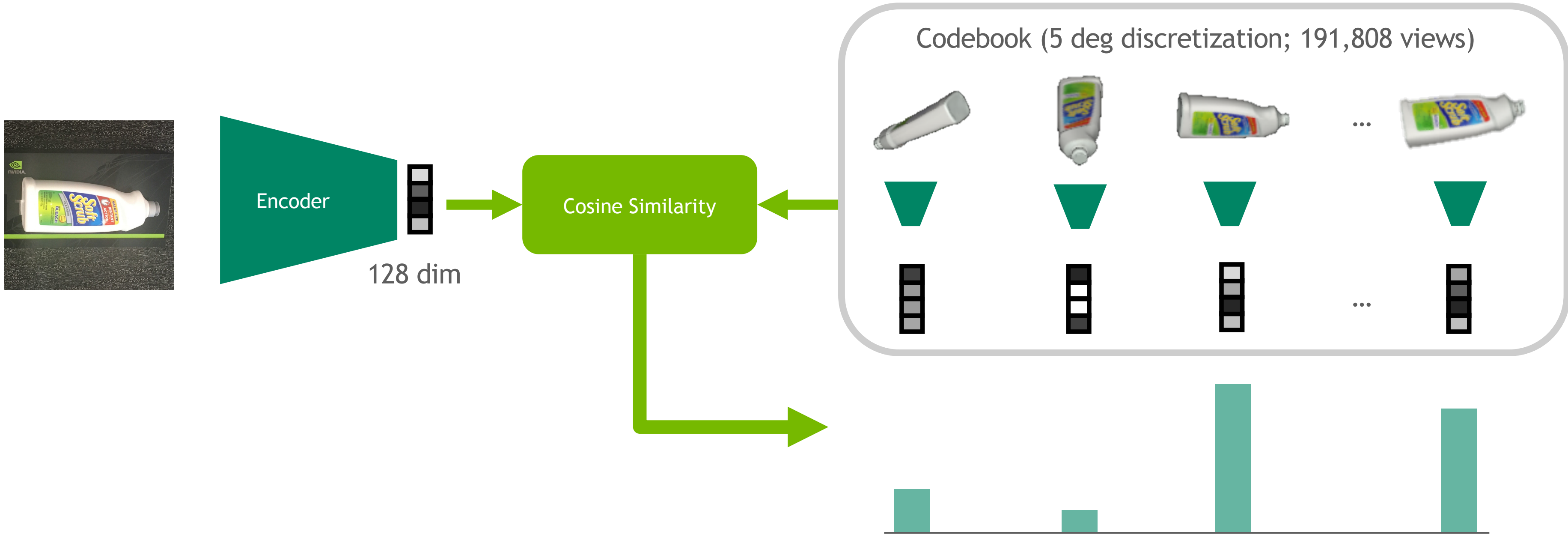
Input view

Target

Bottleneck encodes viewpoint information

TRAINING AN ENCODER IN SIMULATION

Randomly Sample Views onto the Textured Object Model



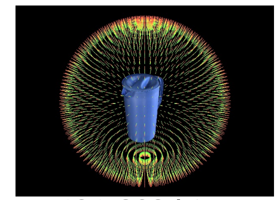
PoseRBPF: 6D PARTICLE FILTER

YCB-Video RGB(-D)
- PoseRBPF:
  ADD: 62.1, ADD-S: 78.4
- PoseCNN:
  ADD: 53.7, ADD-S: 75.9

$$X_i = \{T_i, \mathrm{P}(R_i | T_i, Z_{1:k})\}$$

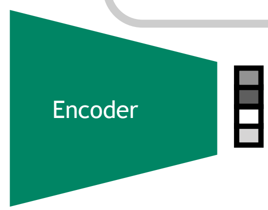
3D Translation
T_i

Orientation Distribution
$\boldsymbol{P(R_i|T_i, Z_{1:k})}$

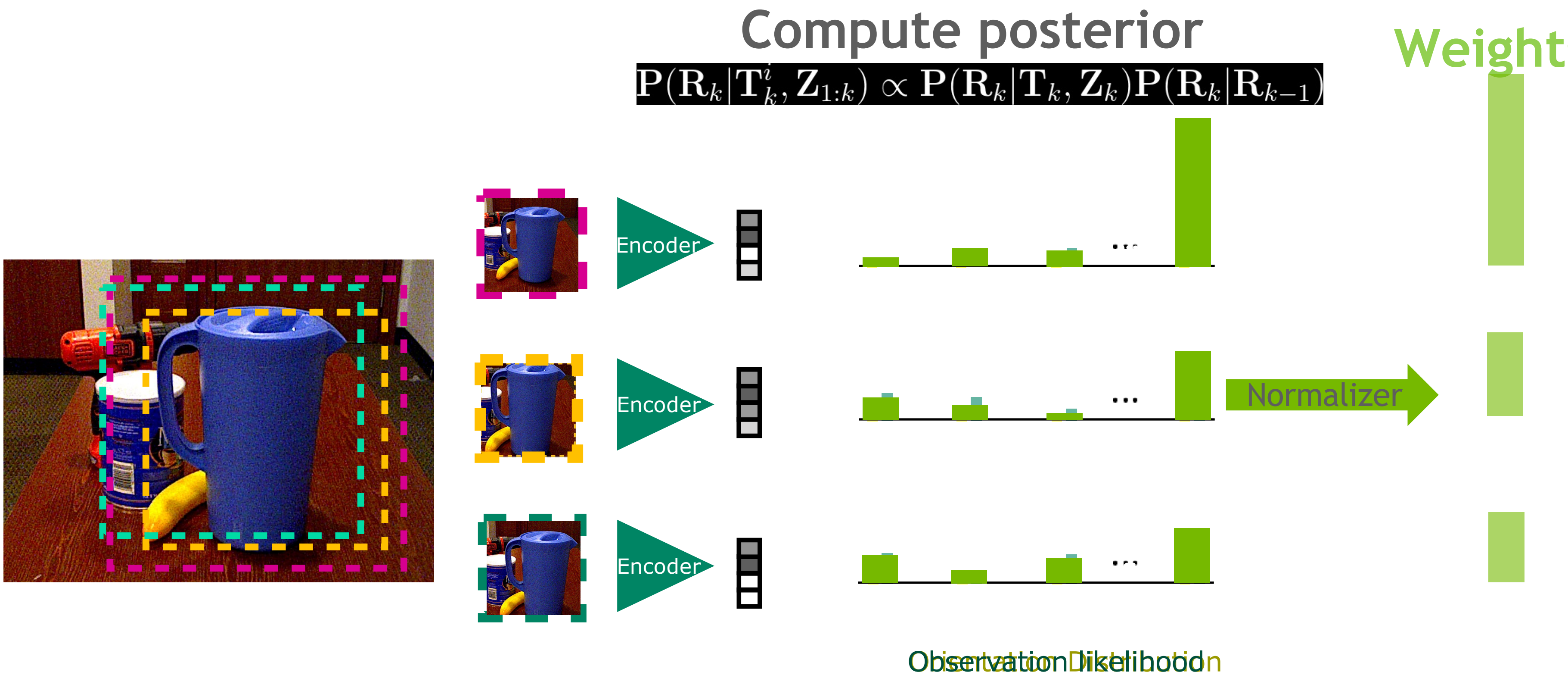


191,808 bins
5 deg resolution

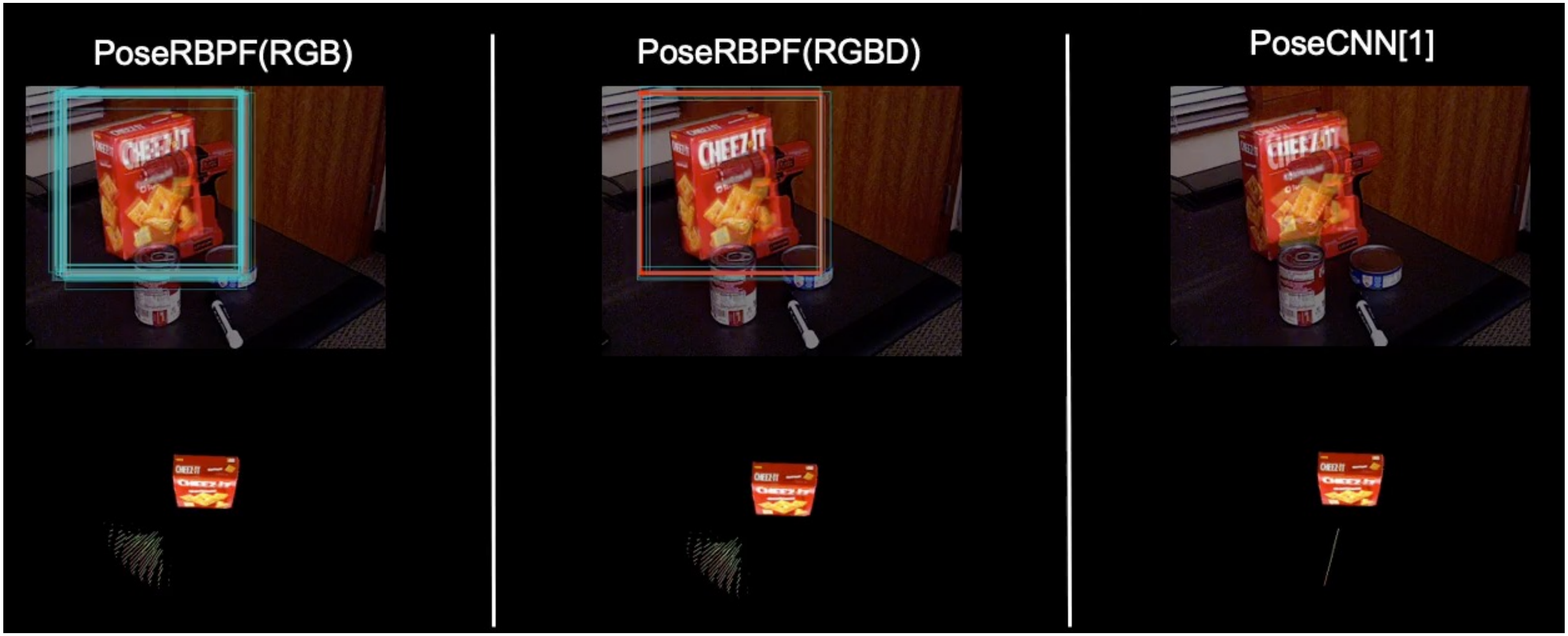
RoI



PoseRBPF: Observation Update



EXAMPLE RESULTS



GLOBAL LOCALIZATION EXAMPLE

Sample Uniformly in Translation Space

1st frame: 5,000 particles, then 500 particles until strong match, then 50 particles
500 particles: 2.6 fps; 50 particles: 20 fps