

CSE-571 Robotics

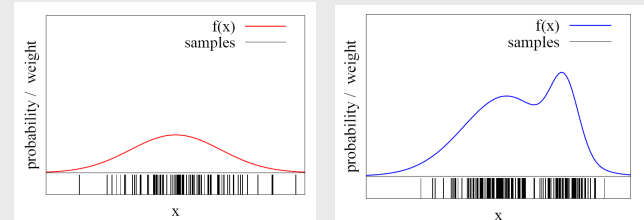
Bayes Filter Implementations

Particle filters

1

Density Approximation

- Particle sets can be used to approximate densities

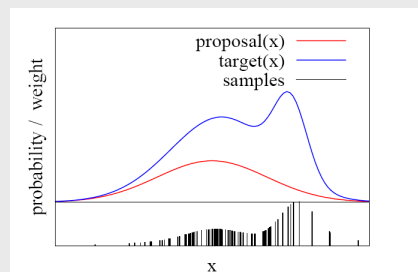


- The more particles fall into an interval, the higher the probability of that interval
- How to draw samples from a function/distribution?

2

Importance Sampling Principle

- We can even use a different distribution g to generate samples from f
- By introducing an importance weight w , we can account for the “differences between g and f ”
- $w = f/g$
- f is often called target
- g is often called proposal



3

3

Bayes Filter Reminder

- Prediction

$$\overline{bel}(x_t) = \int p(x_t | u_t, x_{t-1}) bel(x_{t-1}) dx_{t-1}$$

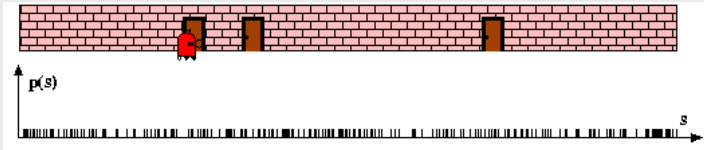
- Correction

$$bel(x_t) = \eta p(z_t | x_t) \overline{bel}(x_t)$$

4

4

Particle Filters

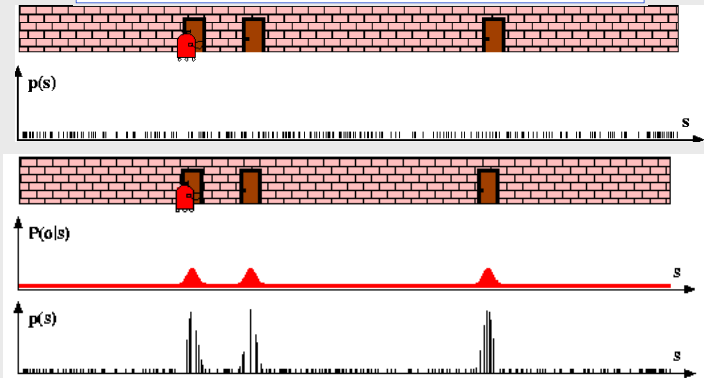


5

Sensor Information: Importance Sampling

$$Bel(x) \leftarrow \alpha p(z|x) Bel^-(x)$$

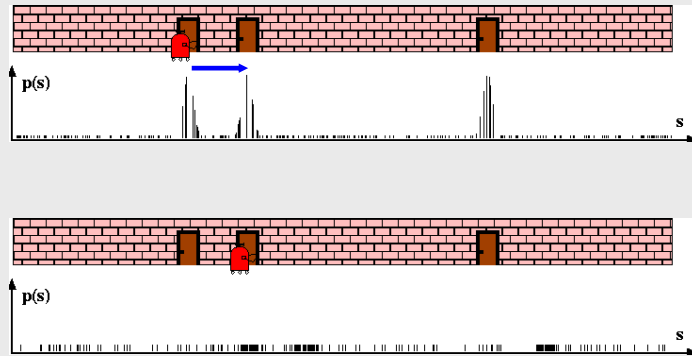
$$w \leftarrow \frac{\alpha p(z|x) Bel^-(x)}{Bel^-(x)} = \alpha p(z|x)$$



6

Robot Motion

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

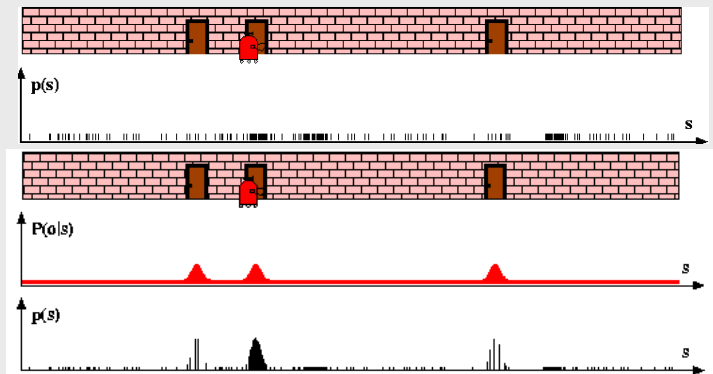


7

Sensor Information: Importance Sampling

$$Bel(x) \leftarrow \alpha p(z|x) Bel^-(x)$$

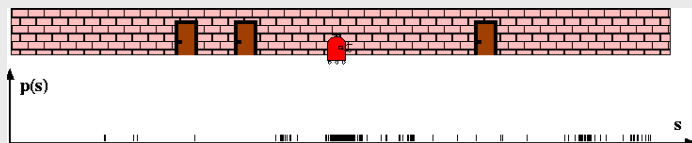
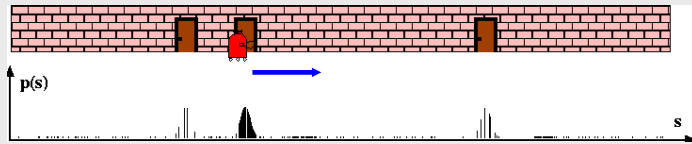
$$w \leftarrow \frac{\alpha p(z|x) Bel^-(x)}{Bel^-(x)} = \alpha p(z|x)$$



8

Robot Motion

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



9

Particle Filter Algorithm

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

draw x_{t-1}^i from $Bel(x_{t-1})$

draw x_t^i from $p(x_t | x_{t-1}^i, u_{t-1})$

Importance factor for x_t^i :

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

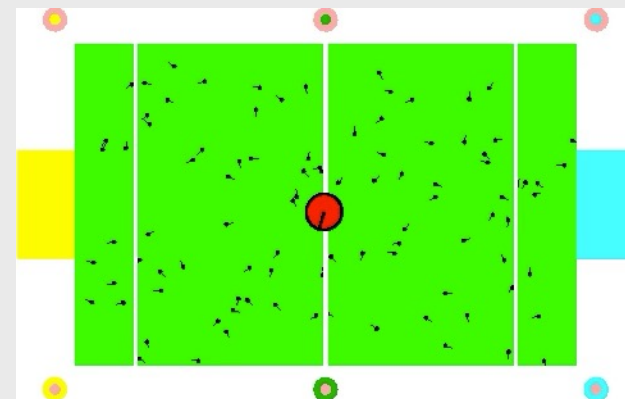
10

Particle Filter Algorithm

1. Algorithm **particle_filter**(S_{t-1}, u_{t-1}, z_t):
2. $S_t = \emptyset, \eta = 0$
3. **For** $i = 1 \dots n$ *Generate new samples*
4. Sample index $j(i)$ from the discrete distribution given by w_{t-1}
5. Sample x_t^i from $p(x_t | x_{t-1}, u_{t-1})$ using $x_{t-1}^{j(i)}$ and u_{t-1}
6. $w_t^i = p(z_t | x_t^i)$ *Compute importance weight*
7. $\eta = \eta + w_t^i$ *Update normalization factor*
8. $S_t = S_t \cup \{ \langle x_t^i, w_t^i \rangle \}$ *Insert*
9. **For** $i = 1 \dots n$
10. $w_t^i = w_t^i / \eta$ *Normalize weights*

11

Localization for AIBO robots



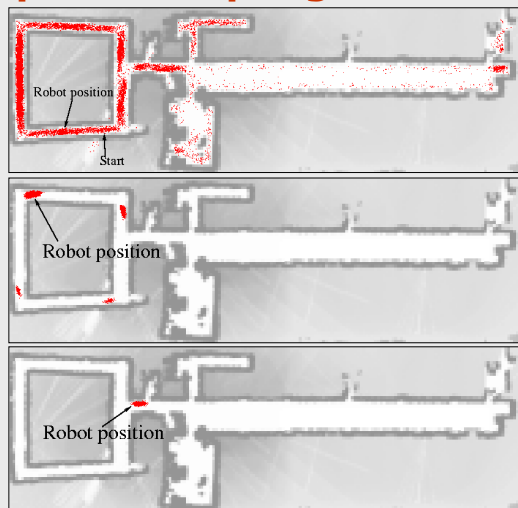
1/14/26

CSE-571 - AI-based Mobile Robotics

12

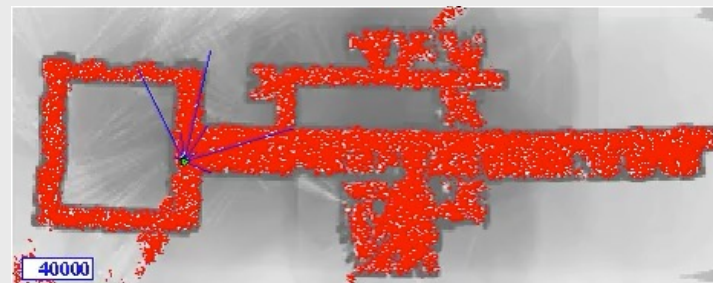
12

Adaptive Sampling



13

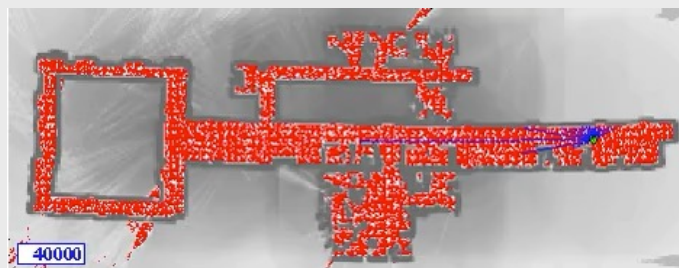
KLD-Sampling Sonar



Adapt number of particles on the fly based on statistical approximation measure

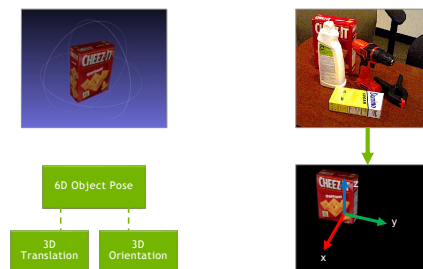
14

KLD-Sampling Laser

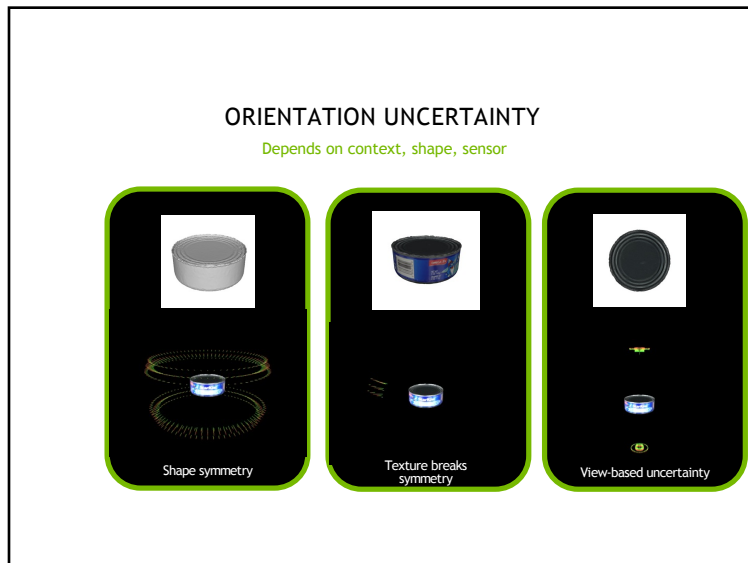


15

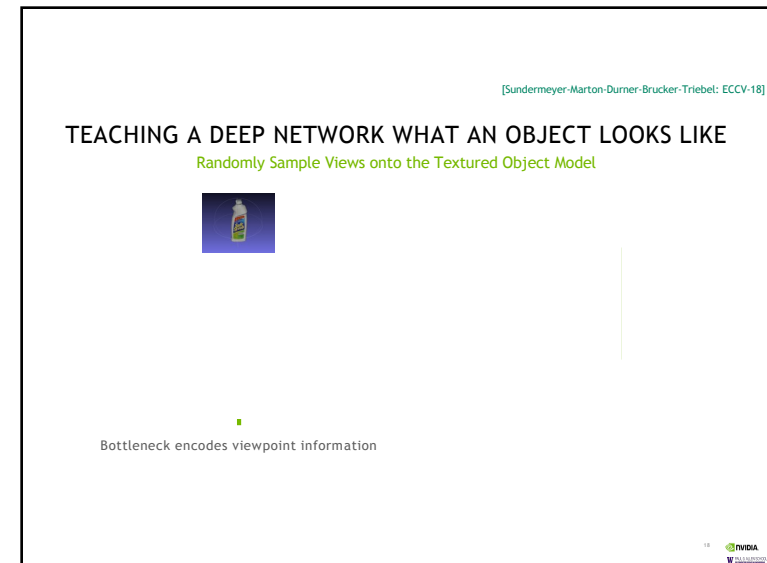
6D OBJECT POSE ESTIMATION



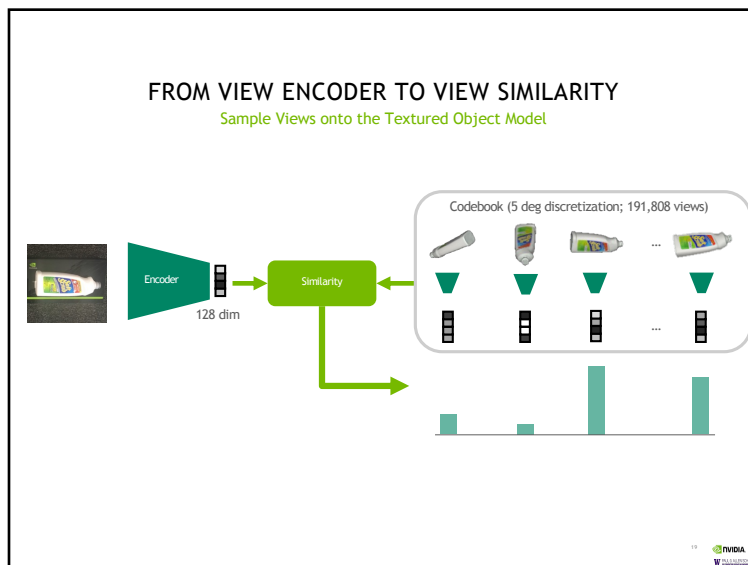
16



17



18



19

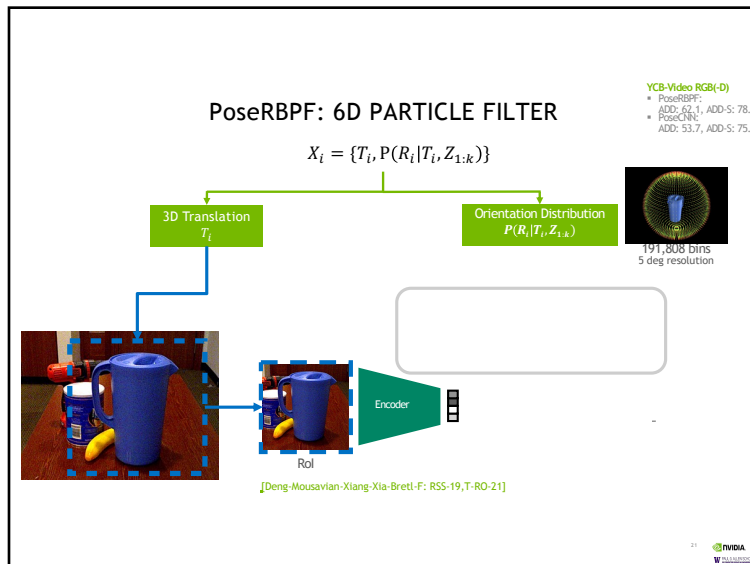
RAO-BLACKWELLIZATION

Factorization to exploit dependencies between variables:

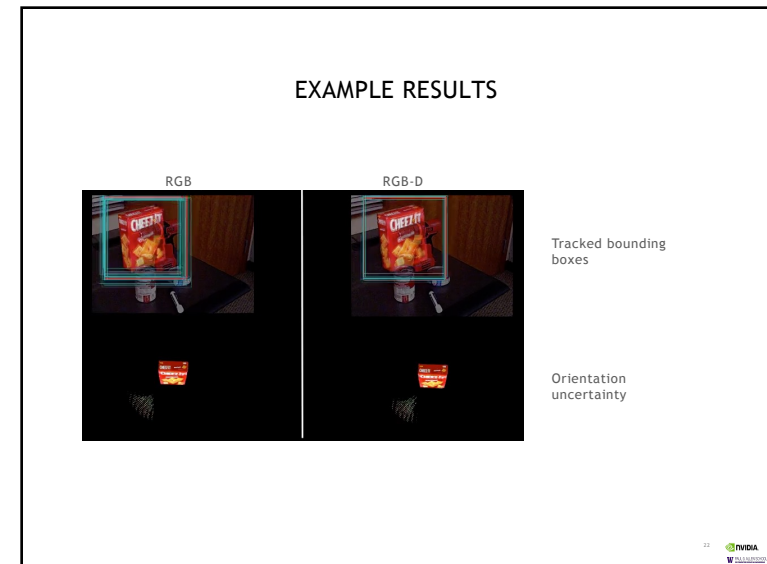
$$p(a, b) = p(b | a) p(a)$$

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

20



21



22



23