Bayes Filter Implementations

Particle filters

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?

Rejection Sampling

- Let us assume that $f(x) \leq 1$ for all $x$
- Sample $x$ from a uniform distribution
- Sample $c$ from $[0,1]$
- If $f(x) > c$ keep the sample otherwise reject the sample

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
**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 $S'$.

**Particle Filters**

**Sensor Information: Importance Sampling**

\[
\begin{align*}
Bel(x) & \leftarrow \alpha \frac{p(z | x) Bel'(x)}{Bel'(x)} \\
w & \leftarrow \alpha \frac{p(z | x) Bel'(x)}{Bel'(x)} = \alpha \frac{p(z | x)}
\end{align*}
\]
Robot Motion

\[ \text{Bel}^t(x) \leftarrow \int p(x \mid u, x') \text{Bel}(x') \, dx' \]

Sensor Information: Importance Sampling

\[ \text{Bel}(x) \leftarrow \alpha \frac{p(z \mid x) \text{Bel}^t(x)}{\text{Bel}^t(x)} = \alpha \frac{p(z \mid x)}{\text{Bel}(x)} \]

Particle Filter Algorithm

\[ \text{Bel}(x_t) = \eta \int \frac{p(z_t \mid x_t) p(x_t \mid x_{t-1}, u_{t-1}) \text{Bel}(x_{t-1})}{p(x_t \mid x_{t-1}, u_{t-1})} \, dx_{t-1} \]

\[ w' = \frac{\text{target distribution}}{\text{proposal distribution}} \]

\[ = \frac{\eta \, p(z_t \mid x_t) \, p(x_t \mid x_{t-1}, u_{t-1}) \, \text{Bel}(x_{t-1})}{p(x_t \mid x_{t-1}, u_{t-1})} \]

\[ \propto p(z_t \mid x_t) \]
Particle Filter Algorithm

1. Algorithm particle_filter( S_{t-1}, u_{t-1}, z_t):
2. $S_0 = \emptyset$, $\eta = 0$
3. For $i = 1...n$
4. Sample index $j(i)$ from the discrete distribution given by $w_{i-1}$
5. Sample $x'_i$ from $p(x'_i | S_{i-1}, u_{i-1})$ using $x^{(i)}_{i-1}$ and $u_{i-1}$
6. $w'_i = p(z | x'_i)$
7. $\eta = \eta + w'_i$
8. $S_i = S_{i-1} \cup \{<x'_i, w'_i>\}$
9. For $i = 1...n$
10. $w'_i = w'_i / \eta$

Generate new samples
Compute importance weight
Update normalization factor
Insert
Normalize weights

Recovery from Failure

Localization for AIBO robots

Adaptive Sampling
KLD-Sampling Sonar

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

6D OBJECT POSE ESTIMATION

ORIENTATION UNCERTAINTY

- Shape symmetry
- Texture breaks symmetry
- View-based uncertainty

Depends on context, shape, sensor
TEACHING A DEEP NETWORK WHAT AN OBJECT LOOKS LIKE

Randomly Sample Views onto the Textured Object Model

Bottleneck encodes viewpoint information

FROM VIEW ENCODER TO VIEW SIMILARITY

Sample Views onto the Textured Object Model

PoseRBPF: 6D PARTICLE FILTER

EXAMPLE RESULTS
GLOBAL LOCALIZATION EXAMPLE
Sample Uniformly in Translation Space

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