Particle Filtering
for Tracking and Localization
Bayesian Inference

Belief before

Data

Belief after
World Knowledge

Sensor Model $P(z|x)$

Most often analytic expression, can be learned
Recap: Bayes Law

\[ P(x|z) \sim L(x;z)P(x) \]

Belief before = \( P(x) \)  
Belief after = \( P(x|z) \)

Prior Distribution of \( x \)  
Likelihood of \( x \) given \( Z \)  
Posterior Distribution of \( x \) given \( Z \)
Example: 1D Robot Localization

Prior $P(x)$

Likelihood $L(x; z)$

Posterior $P(x|z)$
Problem: Large Open Spaces

- Walls and obstacles out of range
- Sonar and laser have problems
- Horizontally mounted sensors have problems
Problem: Large Crowds

- One solution: Robust filtering
Solution: Ceiling Camera

- Upward looking camera
- Model of the world = Ceiling Mosaic
Global Alignment (other talk)
Vision based Sensor

\[ P(z|x) \]

\[ h(x) \]

\[ z \]
Under Light
Next to Light
Various Density Representations

- Gaussian centered around mean $x,y$
- Mixture of Gaussians
- Finite element i.e. histogram
- Does not scale to large state spaces encountered in computer vision & robotics
Hidden Markov Models
Kalman Filter = Very Easy

- Think adding quadratics
- Then Minimize
- Dynamics = Enlarge Quadratic
Kalman Filter

- Very powerful
- Gaussian, unimodal
Example: 2D Robot Location

State space = 2D, infinite #states
Markov Localization

- Fine discretization over \(\{x, y, \theta\}\)
- Very successful: Rhino, Minerva, Xavier…
Dynamic Markov Localization

- Burgard et al., IROS 98
- Idea: use Oct-trees
Sampling as Representation

\[ p(x) \]

\[ x_1 \]

\[ x_2 \]
Sampling Advantages

- Arbitrary densities
- Memory = $O(\#\text{samples})$
- Only in “Typical Set”
- Great visualization tool!

- minus: Approximate
Mean and Variance of a Sample

**Mean**

\[
\mu = \int_x x P(x) \, dx
\]

\[
\mu \approx \frac{1}{R} \sum_{r=1}^{R} x^{(r)}
\]

**Variance (1D)**

\[
\sigma^2 = \int_x (x - \mu)^2 P(x) \, dx
\]

\[
\sigma^2 \approx \frac{1}{R} \sum_{r=1}^{R} (x^{(r)} - \hat{\mu})^2
\]
Inference = Monte Carlo Estimates

- Estimate expectation of any function $f$:

$$E_{P(x)}[f(x)] = \int_x f(x) P(x) d^N x$$

$$E_{P(x)}[f(x)] \approx \frac{1}{R} \sum_{r=1}^{R} f(x^{(r)})$$
Monte Carlo Expected Value

\[ E_{P(x,y)}[\alpha] = \int_x \int_y \tan \left( \frac{y}{x} \right) P(x,y) \, dx \, dy \]

\[ E_{P(x,y)}[\alpha] \approx \frac{1}{R} \sum_{r=1}^{R} \tan \left( \frac{y^{(r)}}{x^{(r)}} \right) \]

Expected angle = 30°
How to Sample?

- Target Density $\pi(x)$
- Assumption: we can evaluate $\pi(x)$ up to an arbitrary multiplicative constant

• Why can’t we just sample from $\pi(x)$??
How to Sample?

- Numerical Recipes in C, Chapter 7
- Transformation method: Gaussians etc…
- Rejection sampling
- Importance sampling
Rejection Sampling

- Target Density $\pi(x)$
- Proposal Density $q(x)$
- $\pi$ and $q$ need only be known up to a factor

Image by MacKay
The Good...
...the Bad...

50% Rejection Rate
...and the Ugly.

70% Rejection Rate
Inference by Rejection Sampling

- $P(\text{measured\_angle}|x,y) = N(\text{predicted\_angle}, 3 \text{ degrees})$

Prior$(x,y)$
Posterior$(x,y|\text{measured\_angle}=20^\circ)$
Importance Sampling

• Good Proposal Density would be: prior!
• Problem:
  – No guaranteed $c$ s.t. $c \, P(x) \geq P(x|z)$ for all $x$
• Idea:
  – sample from $P(x)$
  – give each sample $x^{(r)}$ a importance weight equal to $P(Z|x^{(r)})$
Example Importance Sampling

\{ x^{(r)}, y^{(r)} \sim \text{Prior}(x, y), \ w_r = P(Z|x^{(r)}, y^{(r)}) \}
Importance Sampling

- Sample $x^{(r)}$ from $q(x)$
- $w_r = \pi(x^{(r)}) / q(x^{(r)})$
1D Importance Sampling

![Bar graphs showing 1D Importance Sampling](image)
Example 1

• Learn Color Model
• Implement Rejection Sampling
• Implement Importance Sampling
• Add a spatial prior
A Simple Color Model
Likelihood

\[ P(x, y \mid \text{color}) \propto P(\text{color} \mid x, y)P(x, y) \]

\[ P(\text{color} \mid x, y) = \prod_{c=r,g,b} N(\mu_c, \sigma_c^2) \]
References

- Isard & Blake 98, Condensation -- conditional density propagation for visual tracking
- Dellaert, Fox, Burgard & Thrun 99, Monte Carlo Methods Localization for Mobile Robots
- Khan, Balch & Dellaert 04 A Rao-Blackwellized Particle Filter for EigenTracking
Example: 1D Robot Localization

Prior $P(x)$

Likelihood $L(x;z)$

Posterior $P(x|z)$
1D Importance Sampling
Monte Carlo Approximation of Posterior:

\[ P(X_{t-1}|Z^{t-1}) \leftrightarrow \{X_{t-1}^{(i)}, \pi_{t-1}^{(i)}\}_{i=1}^{N} \]
Bayes Filter and Particle Filter

Recursive Bayes Filter Equation:

\[ P(X_t|Z^t) = k P(Z_t|X_t) \int_{X_{t-1}} P(X_t|X_{t-1}) P(X_{t-1}|Z^{t-1}) \]

Monte Carlo Approximation:

\[ P(X_t|Z^t) \approx k P(Z_t|X_t) \sum_r \pi_{t-1}^{(r)} P(X_t|X_{t-1}^{(r)}) \]
Empirical predictive density = Mixture Model

\[ \pi_t^{(s)} = P(Z_t | X_t^{(s)}) \]

First appeared in 70’s, re-discovered by Kitagawa, Isard, ...
Monte Carlo Localization

\[
S_{k-1} \quad S'_k \quad \text{weighted } S'_k \quad S_k
\]

- Predict
- Weight
- Resample
3D Particle filter for robot pose: Monte Carlo Localization

Dellaert, Fox, Burgard & Thrun ICRA 99
Resampling

• Importance Sampling => weighted
• To get back a fair sample:
  – Resample from the weighted samples according to the importance weights
  – efficient O(N) algorithms exist
Condensation Algorithm

• Sequential Estimation
• Iterates over:
  – Prediction with motion model
  – Importance Sampling for Inference
  – Resampling
1. Prediction Phase

\[ P(x_t | \bullet, u) \]
2. Measurement Phase

\[ P(Z|x_t) \]
3. Resampling Step

\[ O(N) \]
Example 1

- Start from Importance Sampling w Prior
- Implement Sample Mean
- Try increasing nrSamples
- Implement Resampling Step
- Implement Particle Motion Model
A Simple Color Model
Likelihood

\[ P(x, y \mid \text{color}) \propto P(\text{color} \mid x, y)P(x, y) \]

\[ P(\text{color} \mid x, y) = \prod_{c=r,g,b} N(\mu_c, \sigma_c^2) \]