Particle Filtering

*aka Condensation, Sequential Monte Carlo (SMC), . . .*

**Tomáš Svoboda**, svoboda@cmp.felk.cvut.cz

Czech Technical University in Prague, Center for Machine Perception
http://cmp.felk.cvut.cz

**Last update:** April 27, 2015

- density propagation
- importance sampling
- efficient 3D head tracking by particle filter
- 2D tracking
A probabilistic approach to tracking?

The lecture is rather a practitioner introduction.

- At a certain time we need decide about one state (position) of the target object.
- Inner state representation can be arbitrary.
- Let represent the state of the object by probability density.
- We want to estimate the (hidden) density from (observable) measurements.
- Representing of the probability density by particles is one of the effective choices.
A probabilistic approach to tracking?

The lecture is rather a practitioner introduction.

- At a certain time we need decide about one state (position) of the target object.
- Inner state representation can be arbitrary.
- Let represent the state of the object by probability density.
- We want to estimate the (hidden) density from (observable) measurements.
- Representing of the probability density by particles is one of the effective choices.

Particle filter: Particles at the input, measurements, update, . . . , particles at the output.
Particle filter in computer vision

- technique known outside computer vision for long
- popularized under the acronym Condensation in 1996 [4]
- Condensation stands for CONditional DENSity propagATION
- simple, easy to implement, robust . . .
- frequently used in many algorithms
- comprehensive overview [2]
- belongs to Monte Carlo Methods, see chapter 29 [6].
Density propagation

Figure from [1]
Particle filtering

Input: \( S_{t-1} = \{(s_{(t-1)i}, \pi_{(t-1)i})\}, \quad i = 1, 2, \ldots, N \).

Output: \( S_t \) and object state (position) if required

Workflow for time \( t \)

1. Resample data \( S_{t-1} \) by using importance sampling.
2. Predict \( \tilde{s}_{(t)i} \), think about position and velocity model.
3. Uncertainty in the state change \( \rightarrow \) noisify the predicted states.
4. Measure how well the predicted states fit the observation, and update weights \( \pi_t \).
5. If needed compute the mean state (where is the target, actually?).
6. Update the prediction model if used.
One condensation step

\[ s^{(n)}_{k-1}, \pi^{(n)}_{k-1} \]

observation density

\[ s^{(n)}_k, \pi^{(n)}_k \]

predict

measure

\[ 2 \text{ Figure from [1]} \]
Importance sampling

**Input:** set of samples with associated probabilities

**Output:** new set of samples where the frequency depends proportionally on their probabilities
Importance sampling

Cumulative distribution function

video: importance sampling

http://visionbook.felk.cvut.cz
Example: 1–D tracking
Example: 1–D tracking, closer look

- Time step
- State value
- State of the system
- Estimated value
- Samples
- Measurement function
- True state
- Weighted samples
- Estimated state
Application: 3D head tracking in multicamera system
3D head tracking in multicamera system—essentials

Assume calibrated system, $P^j$, and motion segmented projections

- Head modeled as ellipsoid
- **State** comprises position, orientation, velocity vector . . .
- Ellipsoid project as ellipses into cameras
- We **measure** how far are the ellipses from contours

We will go step by step . . .
Ellipsoid and its 2D projection

Quadric surface $Q$

$$X^\top Q X = 0$$

project to a (line) conic

$$C^* = PQ^*P^\top$$

point conic $C$ which is dual to $C^*$

$$u^\top C u = 0$$

Dual matrix:

$$C^* = \det(C)C^{-\top}$$

---

[^3]: Image from [3]
Measurement in (multiple) images

Remember, we can efficiently project outline of the ellipsoid to images.

Distance map

- distance map computed just once per image
- measuring samples is just reading out values from a table
Head 3D tracking — results

**Problem:** 3D position only, no orientation . . .
Learning appearance

- Combines stereo and gradient based localization.
- Explanation of the principle [PDF; www\(^4\)]. More in [7].

3D tracking — including appearance

3D tracking — similarity measure

Oponent colors

\[ a = \frac{1}{2}(R - G) , \quad b = \frac{1}{4}(2B - R - G) , \quad a, b \in \langle -128, 127 \rangle. \]

Histogram of oponent colors

Bhattacharya distance

\[ \text{bhatta}(I, M) = \sum_{k,l} \sqrt{I_{k,l} \cdot M_{k,l}}. \]
3D tracking — Results

video: 3D tracking including orientation

No post-processing, no smoothing applied.
2D tracking — object modeled by color histogram
References


End