Visual Tracking : Particle Filtering.

1 Introduction

This practical course deals with particle filtering algorithms applied into Visual Tracking. All algorithms will be produced using Matlab. Table 1 is an example of a Matlab macro file that reads, processes and displays an image. Table 2 is an example of a Matlab function. The aim of this course is to design, code and analyze a particle filter visual tracking algorithm.

2 Particle Filter

Particle Filter is a stochastic sequential algorithm that estimates the probability distribution at each time of a state sequence, using the previous distribution, a prediction and a likelihood function (observation function). *SIR* (*Sequential Importance Resampling*) filters, the most popular in computer vision, are based on an Importance Sampling step.

Let assume that we have an approximation of $p(\mathbf{X}_{t-1}|\mathbf{Z}_{1:t-1})$, the state posterior probability function at time $t-1$ using N discrete weighted samples $\{\mathbf{X}_{t-1}^n, \pi_{t-1}^n\}_{n=1}^N$:

$$p(\mathbf{X}_{t-1}|\mathbf{Z}_{1:t-1}) \approx \sum_{n=1}^{N} \pi_{t-1}^n \delta(\mathbf{X}_{t-1} - \mathbf{X}_{t-1}^n), \tag{1}$$

where δ is the Kronecker function and π_{t-1}^n is the weight associated to the n^{th} sample, $n \in 1...N$, such as $\sum_{n=1}^{N} \pi_{t-1}^n = 1$. The discrete approximation of Chapman Kolmogorov equation is given by :

$$p(\mathbf{X}_t|\mathbf{Z}_{1:t-1}) \approx \sum_{n=1}^{N} \pi_{t-1}^n p(\mathbf{X}_t|\mathbf{X}_{t-1}^n), \tag{2}$$

where $p(\mathbf{X}_t|\mathbf{X}_{t-1})$ is a probability law that describes the dynamics of the system. The equation (2) defines a set of N weighted particles $p(\mathbf{X}_t|\mathbf{X}_{t-1}^n)$, with weigths π_{t-1}^n. This function produces the probability function of the current state, given the previous observation. It updates the previous posterior probability with a dynamic model. The sequential Bayesian Filter can be written as follow :

$$p(\mathbf{X}_t|\mathbf{Z}_{1:t}) \approx C^{-1} p(\mathbf{Z}_t|\mathbf{X}_t) \sum_{n=1}^{N} \pi_{t-1}^n p(\mathbf{X}_t|\mathbf{X}_{t-1}^n). \tag{3}$$

Some samples have large weights whereas other one have ones close to zero. In order to focus around interesting areas of the state space, a new sampling has to be done. We use an importance sampling algorithm, combined to the dynamic step that uses the proposal law :

$$\mathbf{X}_t^n \sim q(\mathbf{X}_t) = \sum_{n=1}^{N} \pi_{t-1}^n p(\mathbf{X}_t|\mathbf{X}_{t-1}^n) \tag{4}$$

This equation produces, at each time, a new set of unweighted particles. For each sample, $\mathbf{X}_t^n$ The likelihood function is applied : $\pi_t^n = P(\mathbf{Z}_t|\mathbf{X}_t^n)$. A set of weighted samples are then generated $\{\mathbf{X}_t^n, \pi_t^n\}_{n=1}^N$, that approximates $p(\mathbf{X}_t|\mathbf{Z}_{1:t})$, the posterior probability function of the state at time t.

The algorithm *SIR* is illustrated in figure 1. It is divided in three main steps :

- (a) Importance sampling : particles drawn according to their weight from the previous set of particles at $t-1$. This step copies particles with a high weight and removes particles with a low weight. The output of this step is a set of unweighted particles.
- (b) Dynamic prediction : move each particle according to a proposal function (that describes the dynamics of the system to track) $p(\mathbf{X}^*|\mathbf{X})$. When no information is available on the dynamic, a random step can be applied $p(\mathbf{X}^*|\mathbf{X}) = \mathcal{N}(0,\sigma)$, where the value of σ has to be carefully designed.
- (c) Estimation of a weight for each new particle, using the likelihood function of the observation $\mathbf{Z}_t$ at time t, given the sample $\mathbf{X}_t^n$: $\pi_t^n = P(\mathbf{Z}_t|\mathbf{X}_t^n)$

3 Questions

1. You have to complete the file `trackpf.m` in order to program a SIR based particle filter. the state is given by a 2D position into the image reference frame that estimates the gravity center of a moving area. The file is composed by :
   - a function `calcpoids2` which estimates the likelihood associated to a set of particles (the likelihood model is out of the topics of this course).
   - a function `resample` that applies an importance sampling algorithm to a set of particles.

   You will choose a random walk dynamics : each particle is propagated according to a gaussian model with a standard deviation of 10 pixels, centered onto the current position of the particle.
2. You have to display, for several standard deviations (1, 10 et 50) of the random walk, the marginal distribution of the state and discuss the results.
3. You have to display, for several size of the filter (5, 10, 50, 100 particles) the output of the filter.
4. Remove the Importance Resampling step and discuss the resulting behavior of the tacking process.
5. The variable `step` is a sample time factor applied to the video sequence. try to increase this factor and discuss the consequence on the tracking.

```
%%%%%%%%%%%%%%%%%%%%%%%%%
%␣Sample␣of␣Matlab␣macro
%%%%%%%%%%%%%%%%%%%%%%%%%
%
%␣Read␣an␣image
[I,map]=imread('meadownb.jpg');
%%%%%%%%%%%%%%%%%%%%%%%%%
%␣Display␣the␣image
image(I);
%␣pause␣press␣a␣key␣to␣resume
pause␣;
colormap(gray(256));
%%%%%%%%%%%%%%%%%%%%%%%%%
%␣Cast␣the␣image␣in␣double␣to␣modify␣it
Id=double(I);
%%%%%%%%%%%%%%%%%%%%%%%%%
%␣Extract␣a␣Window␣of␣interest␣from␣the␣image
%␣from␣line␣100␣to␣␣300␣and
%␣column␣200␣to␣400
I3=Id(100:300,200:400);
%%%%%%%%%%%%%%%%%%%%%%%%%
%␣Sase␣the␣resulting␣image
imwrite(uint8(I3),gray(256),'toto.jpg','JPEG');
```

Table 1 – Exemple de script matlab.

```
%%%%%%%%%%%%
%␣sample␣of␣Matlab␣function␣foo
%␣with␣three␣input␣parameters:␣a,b,c
%␣and␣two␣output␣parameters␣t1␣and␣t2
%%%%%%%%%%%
%%%%%%%%%%%
%␣Prototype
function␣[t1,t2]=foo(a,b,c)
%
t1=a+b;
t2=b+c;
```

Table 2 – Exemple de fonction matlab.

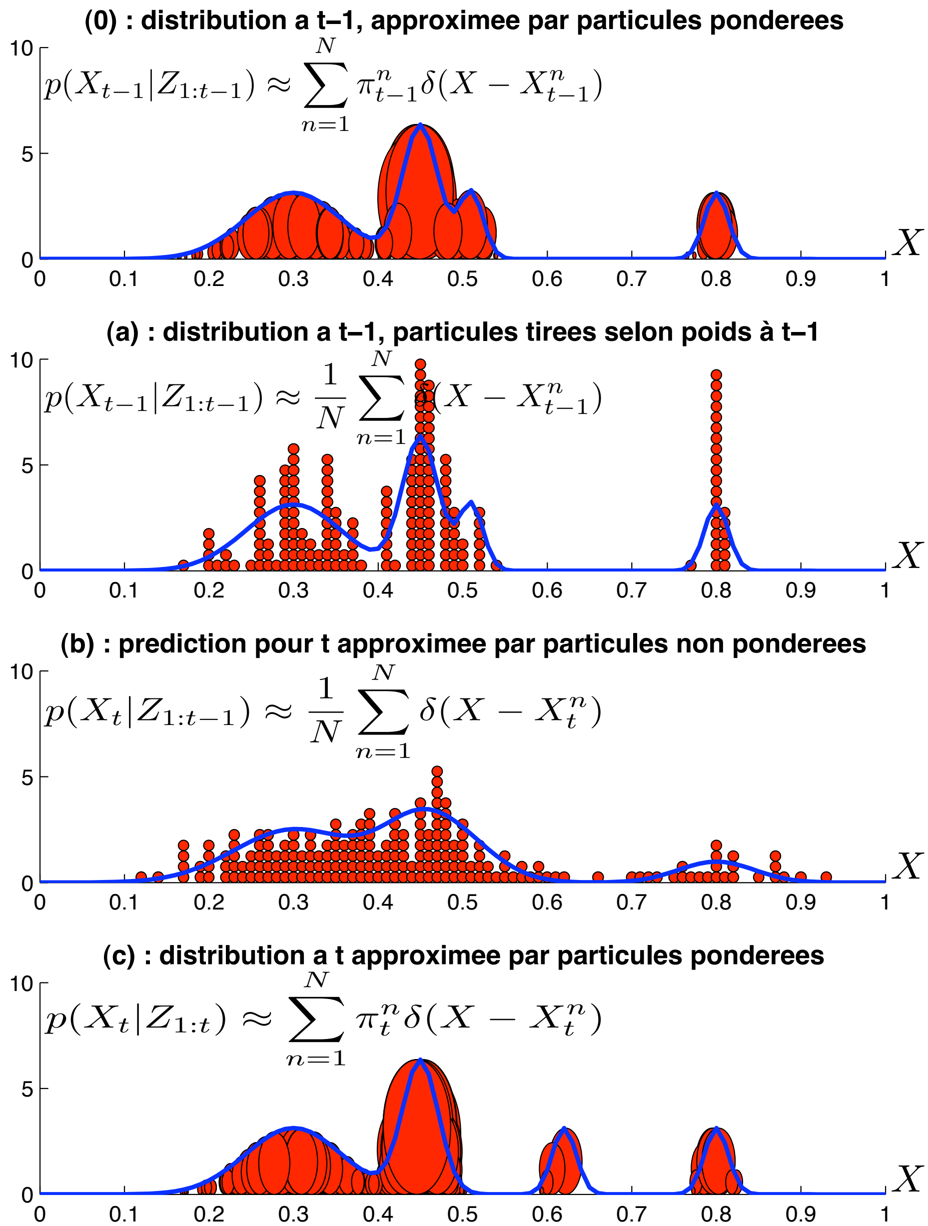

Figure 1 – Illustration of the SIR particle filter algorithm : (a) $t-1$ posterior distribution approximated by a set of weighted particles, (b) $t-1$ posterior distribution approximated by a set of unweighted particles (after the Importance sampling step), and c), t current posterior distribution approximated by a set of weighted particles by the likelihood function.