

AN IMPROVED PARTICLE FILTER ALGORITHM BASED ON NEURAL NETWORK FOR TARGET TRACKING

Qin Wen, Peng Qicong

140 Lab, Institution of Communication and Information Engineering, University of Electronic Science and Technology of China, Chengdu, China

Abstract: To the shortcoming of general particle filter, an improved algorithm based on neural network is proposed and is shown to be more efficient than the general algorithm in the same sample size. The improved algorithm has mainly optimized the choice of importance density. After receiving the samples drawn from prior density, and then adjust the samples with general regression neural network (GRNN), make them approximate the importance density. Apply the new method to target tracking problem, has made the result more precise than the general particle filter.

Key words: particle filter, target tracking, general regression neural network

1. INTRODUCTION

In practice of target tracking, the dynamic system usually is non-linear and non Gaussian. The general method (e.g. Kalman filter) is unable to reach the optimal estimate of target. However, particle filter algorithm based on Bayesian rule uniformly made very good result in state estimate of non-linear and non-Gaussian system. So, it will be a good choice to apply particle filter to the target tracking system.

But, the shortcoming of particle filter can not be ignored either. Because the importance density function is difficult to realize in project practice, so usually adopt prior density and substitute it. This kind of method will reduce

the precision of state estimate. So, people are always studying a better method to approximate the importance density. This paper explores the possibility of using neural network to get optimal result. The below will describe this kind of method and analyze the performance improvement brought by it.

2. PARTICLE FILTER

2.1 Bayesian Rule

The problem of tracking is a process of the state sequence estimated. The dynamics of single state vector at time k is described by a stochastic difference equation

$$x_k = f(x_{k-1}, w_{k-1}) \quad (1)$$

where w_{k-1} is an i.i.d. process noise vector with a known distribution and f is a possibly nonlinear function of the state x_{k-1} . At each discrete time point an observation z_k is obtained, related to the state vector by

$$z_k = h(x_k, v_k) \quad (2)$$

where v_k is an i.i.d. measure noise vector and h is called the measurement function.

Then the state prediction equation

$$p(x_k | z_{1:k-1}) = \int p(x_k | x_{k-1}) p(x_{k-1} | z_{1:k-1}) dx_{k-1} \quad (3)$$

The state update equation

$$p(x_k | z_{1:k}) = \frac{p(z_k | x_k) p(x_k | z_{1:k-1})}{p(z_k | z_{1:k-1})} \quad (4)$$

where the normalizing constant

$$p(z_k | z_{1:k-1}) = \int p(z_k | x_k) p(x_k | z_{1:k-1}) dx_k \quad (5)$$

The above describes the Bayesian estimate rule. Generally, the analytic solution in (3) does exist in some dynamic system. But, it cannot be determined analytically in nonlinear and non-Gaussian system. Therefore, particle filter is a good choice to approximate optimal Bayesian solution.

2.2 Particle Filtering Method

Particle filter is a technique for implementing a recursive Bayesian filter by Monte Carlo simulations. The key idea is to represent the required posterior density function by a set of random samples with associated weights and to compute estimates based on these samples and weights. As the number of samples becomes very large, this MC characterization becomes an equivalent representation to the usual functional description of posterior probability density function, and the SIS (Sequential Importance Sampling) filter approaches the optimal Bayesian estimate [1].

Let $\{x_k^i, w_k^i\}_{i=1}^{N_s}$ denote a random measure that characterizes the posterior density function $p(x_k | z_{1:k})$, where $\{x_k^i, i = 0, L, N_s\}$ is a set of support points with associated weight $\{w_k^i, i = 1, L, N_s\}$. Then, as $N_s \rightarrow \infty$, the posterior density at k can be approximated as

$$p(x_k | z_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(x_k - x_k^i) \quad (6)$$

In the SIS algorithm, the samples x_k^i are drawn from an importance density $q(x_k | x_{k-1}^i, z_{1:k})$, then the weights in (6) update equation can be shown as

$$w_k^i \propto w_{k-1}^i \frac{p(z_k | x_k^i) p(x_k^i | x_{k-1}^i)}{q(x_k^i | x_{k-1}^i, z_k)} \quad (7)$$

In the paper [2], the optimal importance density function that minimizes the variance of the true weights w_k^i conditioned on x_{k-1}^i and z_k has been shown

$$q(x_k | x_{k-1}^i, z_k) = p(x_k | x_{k-1}^i, z_k) \quad (8)$$

The SIS algorithm consists of recursive propagation of the weights and support points as each measurement is received sequentially. A pseudo-code description of this algorithm is give by Algorithm 1.

Algorithm 1 . SIS Particle Filter

```

·for  $i = 1 : N_s$ 
  — draw  $x_k^i : q(x_k | x_{k-1}^i, z_k)$ 
  — assign the particle a weight,  $w_k^i$  according to (7)
·end for
·normalize the weight  $w_k^i = w_k^i / \text{sum}[\{w_k^i\}_{i=1}^{N_s}]$ 
·resample the sample  $w_k^{i*}$ 

```

3. CHOICE OF IMPORTANCE DENSITY BASED ON NEURAL NETWORK

3.1 Shortcoming of SIS

In practice, it is often convenient to choose the importance density to be the prior

$$q(x_k | x_{k-1}^i, z_k) = p(x_k | x_{k-1}^i) \quad (9)$$

Then, substitution of (9) into (7)

$$w_k^i \propto w_{k-1}^i p(z_k | x_k^i) \quad (10)$$

In a situation that the observe precision is low, this method can make better result, but the precision of estimate is not high. Because the current measure value z_k is not considered in importance density function, the samples drawn from importance density and from the real posterior density have greater deviations. Especially when likelihood function is at the end of system state transfer probability density function or measure model has very high precision, the kinds of deviation are more obvious. This can see from Figure1.

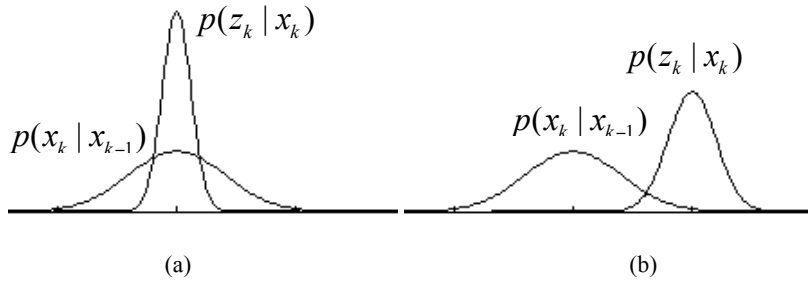


Figure 1. The prior density and likelihood function. (a) The likelihood function is peak. (b) The likelihood function is at the end of prior density function.

To the shortcoming of SIS algorithm, A.Doucet proposes to construct suboptimal importance density to the optimal importance density by using local linearization techniques [2]. R van der Merwe, A.Doucet, etc. propose to estimate a Gaussian approximation to importance density using the unscented transform [3]. Yuan Zejian and Zheng Nanning, etc. propose using Gauss-Hermite filter to sample and reconstruct the importance density in [4]. Peter Torma, etc. propose a local search method to adjust samples, make it more approximate importance density function [5].

This paper applies artificial neural network to the choice of importance density. Using general regression neural network (GRNN) to adjust samples after prediction step can get good result, make samples accord with the posterior density further.

3.2 GRNN

GRNN is a novel neural network proposed by Donald F.Specht in 1991[6]. The basic theory is nonlinear regression analysis. GRNN is different with the BP network. The traditional BP neural network is one kind of typical universal approximation network. One or more weights are influence to each output in network. It causes study speed to be slower; moreover, the weight determined is stochastic, which causes the relationship between input and output after each step of training unstable and the forecast result deviation. The GRNN only needs a simple smooth parameter, does not need to carry on the training process circularly, and does not adjust weight between the neurons in the training process. The network is steady and the computation speed is quick. Therefore, in the real-time target tracking application, GRNN has the superiority compared to BP.

What show in Figure 2 is a feed forward network that can be used to estimate a vector Y from a measurement vector X .

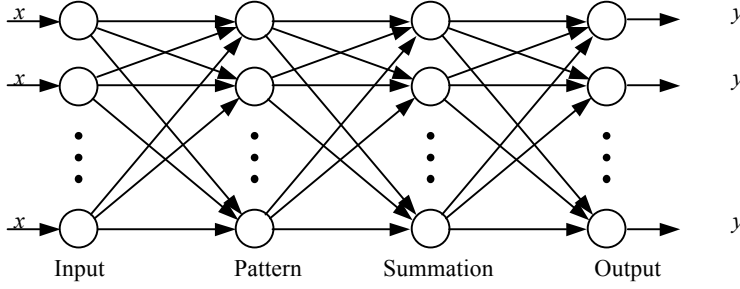


Figure2. GRNN block diagram

Let X be a particular measured value of random variable x , and X^i is sample value of x .

Defining the scalar function

$$D_i^2 = (X - X^i)^T (X - X^i) \quad (11)$$

and performing the indicated integrations yields the following

$$\hat{Y}(X) = \frac{\sum_{i=1}^n Y^i \exp(-\frac{D_i^2}{2\sigma^2})}{\sum_{i=1}^n \exp(-\frac{D_i^2}{2\sigma^2})} \quad (12)$$

where Y^i is sample value of Y . σ is the width of sample probability for each sample X^i and Y^i .

GRNN can be used to adjust samples in particle filter algorithm according to measurement value z_k .

Because when in a concrete target tracking scene, the profile of likelihood function is fixed at any time of k , but its mathematic expression is not clear, only can be obtained by some separate observed values through image processing methods. So we may train the network according to the observed value and make it approach the likelihood function, and then utilizes this network to any samples to carry on the adjustment.

Firstly, the input vector and object vector should be structured to train the network. A group of samples are obtained by sampling equal-space likelihood function. n neighborhood samples and their likelihood function values constitute the input vector and object vector respectively. The

dimension of input vector n and the number of study sample m determine the network configuration: $n \times m \times (n+1) \times n$.

After the network is trained, the samples in particle filter algorithm should be adjusted in form of input vector, and then it can be apply to the network. Firstly, construct a n dimension vector $X_k^i = [x_k^i, x_k^i \pm j\Delta]$, $j\Delta < L$, ($j= 1, \dots, n/2$). The parameter L defines the adjustment range. Then, transform $X_k^i \rightarrow h(X_k^i) - z_k$ as the input vector of GRNN trained.

Finally, through the indication of output vector of network, the sample x_k^i is substituted by optimal point $x_k^i \pm j\Delta$. A series of samples adjusted are more optimally approximate the importance density.

4. SIMULATION

Let X_k represent the state variable at time k , corresponding to the position of the target in the state space

$$X_k = \begin{bmatrix} x_k & y_k \end{bmatrix}^T$$

The two-dimensional target dynamics is given by [5]

$$\begin{aligned} \hat{X}_{k+1} &= X_k + S_{k+1}\Delta_k + W_k \\ S_{k+1} &= (2B_{k+1} - 1)S_k \\ U_{k+1} &= \mathcal{X}(|\hat{X}_{k+1}| \leq K) \\ X_{k+1} &= U_{k+1}\hat{X}_{k+1} + (1 - U_{k+1})X_k \\ \Delta_{k+1} &= U_{k+1}(X_{k+1} - X_k) + (1 - U_{k+1})(X_{k+1} - \hat{X}_{k+1}) \end{aligned}$$

where $W_k \sim N(0, \sigma^2)$ are i.i.d. Gaussian random variables, and B_k is a Bernoulli variable.

The measure model is $Z_k = X_k + V_k$, where $V_k \sim N(0, \delta^2)$ i.i.d. The dynamics is highly nonlinear.

We test the general SIS algorithm and the improved algorithm based on GRNN to this dynamics model respectively. Figure 3 shows a typical sequence of tracking a ball. The number of particles was chosen to be as 100 in two kinds of algorithms. The dimension of input vector in GRNN is 7. Obviously, the two algorithms all find the position of ball precisely along

with the movement of target. However, the improved algorithm has higher accuracy than general one. Figure 4 shows the tracking precision as a function of the time. The improved algorithm has the obvious superiority compared to the general particle filter algorithm at the average deviation pixels.

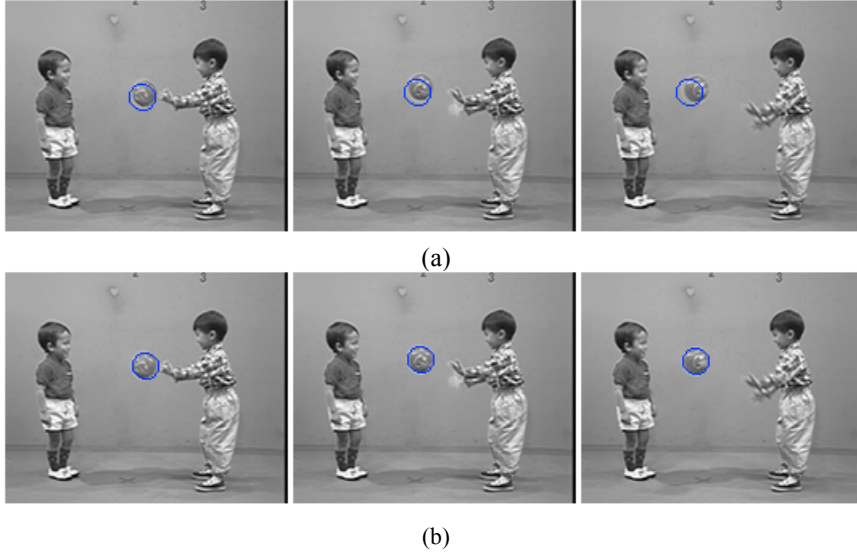


Figure 3. The result of simulation. (a) The result of tracking with the general SIS algorithm. (b) The result of tracking with the improved algorithm based on GRNN adjustment

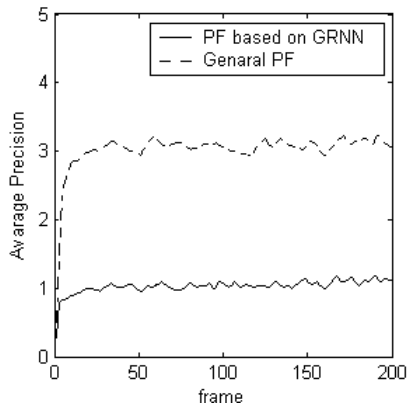


Figure 4. Tracking precision as a function of time

5. CONCLUSION

In this article, an improved particle filter algorithm based on GRNN is proposed to solve the problem of importance density function choosing. The

new method is shown to improve the performance of samples and to increase robustness as compared with the previous method proposed, whilst the novel algorithm minimizes the expected distortion in the configuration space.

One weakness of the approach is that the GRNN will spend more time than general particle filter algorithm. However, the price of time is worth. Moreover, it is in the range that the system can bear.

In conclusion, we believe that it is a significant exploration to adopt neural network to solve the optimal importance density. It will be a valuable step towards the implementation of highly efficient tracking.

REFERENCES

1. M.Sanjeev Arulampalam, Simon Maskell, Neil Gordon. A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking. *IEEE Transactions on Signal Processing*, 2002,50(2):174~188
2. A.Doucet. On sequential Monte Carlo methods for Bayesian filtering. Dept. Eng., Univ. Cambridge, UK, Tech. Rep., 1998.
3. R.van der Merwe, A.Doucet, N.de Freitas, E.Wan. The Unscented Particle Filter. *Adv. Neural Inform. Process. Syst.*, Dec.2000
4. Yuan Zejian, Zheng Nanning, Jia Xinchun. The Gauss-Hermite Particle Filter. *Dian Zi Xue Bao*, 2003,31(7):970~973
5. Peter Torma, Csaba Szepesvari. LS-N-IPS:an improvement of particle filters by means of local search. In *Proc. Non-Linear Control Systems (NOLCOS'01)*,2001
6. Specht D F. A General Regression Neural Network. *IEEE Transactions on Neural Networks*, 1991, 2 (6):568~576