Target Tracking Based on Optimized Particle Filter Algorithm

Junying Meng[1, 2], Jiaomin Liu [1], Juan Wang[1], Ming Han[1]
[1]College of Information Science and Engineering, Yanshan University, Qinhuangdao, China
[2]Department of Computer science, Shijiazhuang University, Shijiazhuang, China
Email:daishan74@126.com

Abstract—**Particle filter is a probability estimation method based on Bayesian framework and it has unique advantage to describe the target tracking non-linear and non-Gaussian. In this paper, Firstly, analyses the particle degeneracy and sample impoverishment in particle filter multi-target tracking algorithm, and secondly, it applies Markov Chain Monte Carlo (MCMC) method to improve re-sampling process and enhance performance of particle filter algorithm. Finally, the performance of the proposed method is certificated by experiment that tracking multiple targets of similar appearance and complex motion. The results show the efficacy of the proposed method in multi-target tracking.**

Index Terms—**particle filter; multi-target tracking; sequential important sampling; MCMC**

I. Introduction

Target tracking is an important element of surveillance, guidance, or obstacle avoidance systems, whose main idea to determine the number, position, and movement of targets. The fundamental building block of a tracking system is a filter for recursive target-state estimation.

In recent years, many single-target video sequence tracking system are successfully developed one after another. If it is a linear Gaussian state-space model, is possible to derive an exact analytical expression to compute the evolving sequence of posterior distributions. This recursion is the well-known and widespread Kalman filter. If the data modeled as a partially observed, finite state-space Markov chain, it is also possible to obtain an analytical solution, which is known as the hidden Markov model HMM filter. These two filters are the most ubiquitous and famous ones, yet there are a few other dynamic systems that admit finite dimensional filters. And there are more classic algorithms that widely applied are: Nearest Neighbor Filter (NNF)[1], Joint Probability Data Association Filter (JPDAF)[2] and Multiple Hypothesis Tracking (MHT)[3], etc..

The aforementioned filters rely on various assumptions to ensure mathematical tractability. However, Real data can be very complex, typically involving elements of non-Gaussianity, high dimensionality and nonlinearity, which conditions usually preclude analytic solution. This is a problem of fundamental importance permeates most disciplines of science. According to the field of interest, the problem appears under many different names, including Bayesian filtering, optimal filtering, stochastic filtering and on-line inference and learning. For over thirty years, many approximation schemes, such as the extended Kalman filter, Gaussian sum approximations and grid-based filters, have been proposed to surmount this problem. The first two methods fail to take into account all the salient statistical features of the processes under consideration, leading quite often to poor results. Grid-based filters, base on deterministic numerical integration methods, can lead to accurate results, but are difficult to implement and too computationally expensive to be of any practical use in high dimensions[4].

Particle filters (PF) methods are a set of simulation-based methods, which provide a convenient and attractive approach to computing the posterior distributions. Unlike grid-based methods, PF methods are very flexible, easy to implement, parallelizable and applicable in very general settings. The advent of cheap and formidable computational power, in conjunction with some recent developments in applied statistics, engineering and probability, have stimulated many advancements in this field.

With particle filters putting into application, the problems existed in classical algorithms above said have been improved. Particle filtering (PF) technique is a Optimal Regression Bayesian filtering algorithm based on MCMC simulation, which is not limited by linear error and high Gaussian noise hypothesis and is applicable for non-linear non-Gaussian model. The basic idea is to take a series of weighted particles from current system state distribution to estimate and update the next system state.

In this paper, we analyses the particle degeneracy and sample impoverishment in particle filter multi-target tracking algorithm, Applies Markov Chain Monte Carlo (MCMC) method to improve re-sampling process and enhance performance of particle filter algorithm. Through experiments, we tested the performance of the proposed method that tracking multiple targets of similar appearance and complex motion. From the result it can be seen, this method have the outstanding effect in multi-target tracking.

doi:10.4304/jsw.8.5.1140-1144

II. Particle Filter Theory and Application

Particle filter is a probability estimation method based on Bayesian framework and it is very suitable to describe the target tracking uncertainty. The essence is to realize Bayesian filter in non-parametric Monte Carlo simulation method.

Particle filter approach itself is able to express a number of assumptions in particle sets, so it can be used to solve multi-target tracking problem. Due to data association is only considered in a given period of time, the complex of data association is thus reduced. Using hybrid bootstrap filter to solve the data association problem, in which each particle involves single target state information and expresses one target state hypothesis; using Gaussian mixer model to express posterior distribution of all targets under the given observation conditions, and each model of posterior distribution corresponds to a target[6]. However, Gaussian mixer model expressing posterior distribution will lead to target lost in occlusion, because as the tracking keeps on, the weight of the particle expressing occluded target will become very small, and the particle will be dropped in re-sampling process.

The core idea of particle filter algorithm is to use weighting of a series of random samples and posterior probability density required by expression, to get the estimated state value. When the sample number is very large, such probability estimation will be equal to posterior probability density. Assume Ns indicate the particle number, then $\{X_k^i, i=1,...,N_s\}$ means a support point set, and its corresponding weight is $\{w_k^i, i=1,...,N_s\}$, and normalized weight is $\sum_{i=1}^{N_s} w_k^i = 1$, then $\{X_k^i, w_k^i\}_{i=1}^{N_s}$ indicates the random particle set describing posterior density. Thereupon, posterior probability density at the time k can use discrete weight sum that is approximate to:

$$p(X_k|Z_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(X_k - X_k^i) \tag{1}$$

In which, the weight w_k^i can be sampled and selected from important density function $q(X_k|Z_{1:k})$ in sequential important sampling method.

If the sample X_k^i can be obtained from important density $q(X_k|Z_{1:k})$, then the weight of the i'th particle can be defined as:

$$w_k^i \propto \frac{p(X_k^i|Z_{1:k})}{q(X_k^i|Z_{1:k})} \tag{2}$$

If the important density function can be decomposed as follows:

$$q(X_k|Z_{1:k}) = q(X_k|X_{k-1}, Z_{1:k})q(X_{k-1}|Z_{1:k-1}) \tag{3}$$

Then the posterior probability density can be expressed as:

$$\begin{aligned} p(X_k|Z_{1:k}) &= \frac{p(Z_k|X_k)p(X_k|Z_{1:k-1})}{p(Z_k|Z_{1:k-1})} \\ &= \frac{p(Z_k|X_k)p(X_k|X_{k-1}, Z_{1:k-1})p(X_{k-1}|Z_{1:k-1})}{p(Z_k|Z_{1:k-1})} \\ &= \frac{p(Z_k|X_k)p(X_k|X_{k-1})}{p(Z_k|Z_{1:k-1})} p(X_{k-1}|Z_{1:k-1}) \\ &\propto p(Z_k|X_k)p(X_k|X_{k-1})p(X_{k-1}|Z_{1:k-1}) \end{aligned} \tag{4}$$

And updated formula for weights is obtained therefore:

$$\begin{aligned} w_k^i &\propto \frac{p(Z_k|X_k^i)p(X_k^i|X_{k-1}^i)p(X_{k-1}^i|Z_{1:k-1})}{q(X_k^i|X_{k-1}^i, Z_{1:k})q(X_{k-1}^i|Z_{1:k-1})} \\ &= w_{k-1}^i \frac{p(Z_k|X_k^i)p(X_k^i|X_{k-1}^i)}{q(X_k^i|X_{k-1}^i, Z_{1:k})} \end{aligned} \tag{5}$$

Weights can be normalized as:

$$\tilde{w}_k^i = \frac{w_k^i}{\sum_{i=1}^{N_s} w_k^i} \tag{6}$$

If $q(X_k|X_{k-1}, Z_{1:k}) = q(X_k|X_{k-1}, Z_k)$ is achieved, namely the important density function only depends on X_{k-1} and Z_k, then only storage sample X_k^i but not X_{k-1}^i and the past observation $Z_{1:k-1}$ is needed, therefore computation storage can be greatly reduced. At this time the weight is revised as:

$$w_k^i \propto w_{k-1}^i \frac{p(Z_k|X_k^i)p(X_k^i|X_{k-1}^i)}{q(X_k^i|X_{k-1}^i, Z_k)} \tag{7}$$

Thus, the posterior probability density at the time K can use discrete weight sum that approximate to：

$$p(X_k|Z_{1:k}) \approx \sum_{i=1}^{N_s} \tilde{w}_k^i \delta(X_k - X_k^i) \tag{8}$$

Therefore, particle filter algorithm is to obtain samples mainly from important density function, get corresponding weight in iteration as successive arrival of measured values, and finally represent the posterior probability density in the form of weight sum, and get the estimated state value.

For multi-target tracking system, N (quantity) particles are involved in initial particle set $S_0 = \left(s_0^n, \frac{1}{N}\right)_{n=1,...,N}$, in which each element $s_0^{n,i}$ from $i=1,...,M$ is obtained from independent $p(X_0^i)$ sampling. The particle set at the time t-1 is assumed as $S_{t-1} = (s_{t-1}^n, p_{t-1}^n)_{n=1,...,N}$, in which $\sum_{n=1}^N p_{t-1}^n = 1$. Each particle is a vector of dimension $\sum_{i=1}^M n_x^i$, and $s_{t-i}^{n,i}$ represents the i'th element in s_{t-1}^n, and n_x^i represents the state vector dimension of the i'th target.

Each iteration in particle filter algorithm is divided into two steps: prediction and weight updating. Prediction means sampling from proposed density function F_t^i, and proposed density function is consistent with the target motion model; weight updating is to make the weight at the time t-1 multiplied by the observation likelihood.

$$\tilde{s}_t^{\,n} = \begin{pmatrix} F_t^{\,i}\left(s_{t-1}^{n,1}, v_t^{n,M}\right) \\ \vdots \\ F_t^{\,M}\left(s_{t-1}^{n,M}, v_t^{n,M}\right) \end{pmatrix} \quad (9)$$

For the likelihood calculation of the nth particle, the observed value $\tilde{s}_t^{\,n}, n = 1,..., N$ can be expressed as:

$$p\left(Z_t = \left(z_t^1,..., z_t^{m_t}\right) \middle| X_t = \tilde{s}_t^{\,n}\right) = \prod_{j=1}^{m_t} p\left(z_t^j \middle| \tilde{s}_t^{\,n}\right)$$

$$\propto \prod_{j=1}^{m_t}\left[\frac{q_t^0}{V} + \sum_{i=1}^{M} l_t^i\left(z_t^j; \tilde{s}_t^{\,n,i}\right) q_t^i\right] \quad (10)$$

In which, $l_t^i\left(z_t^j; \tilde{s}_t^{\,n,i}\right) = p\left(z_t^j \middle| K_t^j = i, X_t^i = \tilde{s}_t^{\,n,i}\right)$, $q_t^i, i = 1,..., M^t$ means the probability of the j'th observed value from the i'th target, and M^t means the quantity of target at the time t.

III. Particle Degeneracy and Re-sampling

The basic problem to be solved in sequential importance sampling algorithm is particle degeneracy, after a few or multiple recursion, the weights of most particles become very small and only a few particles have a relatively large weights. Degeneracy is inevitable: the variance of particle weight will grow continuously over time. Since the small weights of particles contribute little to problem solution, a great deal of computing power will be wasted on them if the recursion continues, and it is obviously unnecessary.

Re-sampling technique is used herein to solve particle degeneracy, namely removing the particles of small weight and reproducing those of large weights[7]. Detailed process is as follows.

After systematic observation, the first step is to recalculate and confirm the weight ranges of the particles. The realistic particles will be granted relatively large weights and those deviating from reality will be given relatively small ones.

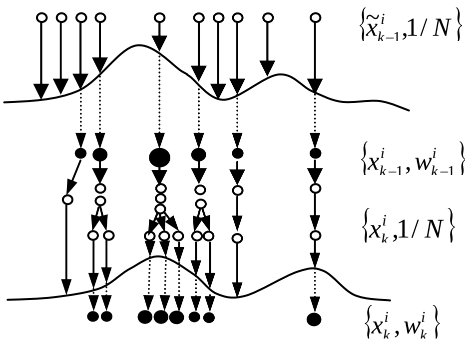

Figure 1.The main process of re-sampling

The second step is re-sampling process, in which the particles of large weights will derive much more "offspring" particles and those of small weights will correspondingly derive less ones, moreover, the weights of "offspring" particles will be re-set.

The third step is system state transition process, in which the state of each particle at the time t will be predicted through adding a random amount of particles.

The forth step is system observation process at time t, similar to the first step, the final representation of target state will be obtained through weighting of a numbers of particles.

These new particles propagated into the calculate of next frame, Then the dynamic model change the position of particles and the observation model change the weight of particles, determine the target position. Re-sampling cycling constantly, this process shown in Figure 1.

IV. Strategy for Sampling Improvement

Particle filter re-sampling inhibits the weight degeneracy, but also introduces other problems. At first, the particles are no longer independent, reducing the opportunity of parallel computing because of continuous re-definition of new particle set. Second, the particles of relatively large weights will be chosen for many times, weakening the particle diversity, and the sample particles contain many duplicate points, when the system noise is small, the said will be obvious, and after several iteration, all particles will converge to a point, and this is known as particle depletion.

Particle depletion resulted from re-sampling process makes the number of particles expressing PDF state too small and therefore inadequate, while unlimited increase of particle number is not realistic.

Markov chain Monte Carlo (MCMC) method is introduced to generate samples from target distribution through constructing Markov chain, which has a good convergence effect[8]. In each process of iteration of sequential important sampling, the particles can move to different places by combining with MCMC, so that particle depletion is avoided, and furthermore, Markov chain can push the particles to the places closer to PDF state and make the sample distribution more reasonable. There are many MCMC methods put into application, and MetropolisHasting method is adopted herein.

Specific re-sampling process is as follows:

According to the samples uniformly distributed in the range (0,1), thresholds u~U(0,1) are obtained;

Sampling $x_t^{*(i)}$ as per distribution probability $p(x_t \mid x_{t-1}^{(i)})$, i.e. $x_t^{*(i)} \sim p(x_t \mid x_{t-1}^{(i)})$;

Accept $x_t^{*(i)}$,if $u < \min[1, \frac{p(y_t \mid x_t^{*(i)})}{p(y_t \mid \tilde{x}_t^{(i)})}]$; otherwise, drop $x_t^{*(i)}$, make $x_t^{(i)} = \tilde{x}_t^{(i)}$.

V. Template Updating

Selection of target template is an important part of visual tracking algorithm, and a good target template shall be distinctive and unique to ensure the tracking accuracy and effectiveness. In motion process, the targets will be changed due to effects of its motion, light and perspective, and only appropriately and reasonably updating of target template can overcome to some extent the impaction of such changes on tracking effect[9]. Reasonable update strategy shall be able to adapt to slow changes of target characteristics, but also rapid changes.

Template is generally divided into stationary template and dynamic template. Stationary template is often applied because of sample and reliable. However, characteristics of moving target will be changed over time, when the change of moving target state leads to corresponding change of its characteristic, it requires the algorithm to take appropriate strategy to response, and obviously the stationary template can not satisfy such requirement.

Dynamic template is a resolution responding to the requirements above said. The simplest update rule for dynamic template is update frame by frame, which abandons all previous template information and adopts the best-matched sub-region image of previous time as current target template. However, Due to the influence of the shadow and the changes of light, deformation or accumulation of matching error, the dynamic template will easily lead to target tracking drift and even lost.

Dynamic template can be expressed as a forgetting process as follows:

$$M_{updated} = \alpha \cdot M_{fixed} + (1-\alpha) M_{new} \qquad (11)$$

In which, α indicates retention of stationary template that can takes empirical value. Coefficient Bhattacharyya indicating target similarity is adopted herein as a parameter, compared with empirical value, it is more in line with update requirement. M_{fixed} indicates dynamic template, and generally it is target weighted color histogram in initial position. $M_{updated}$ indicates new template, and generally it is target weighted color histogram in estimated position. By using the dynamic template update rule above said, the target weighted color histogram model contains the target color information of initial time and current time, but also makes real-time adjustment of update rate according to target similarity in estimated position, which can effectively inhibit tracking errors from accumulating and tracking target from drifting.

VI. Experiments and result

We test the MCMC method in three different video images. In Figure 2, there was the shadow interference region. The two target walk form the bright area to dark area. In traditional method, this situation will make the tracking failed. In Figure 3, the targets sometime overlapped. How to identify the overlapped targets is the aim of this experiment. Figure 4 is a traffic video. These two vehicles have the similar appearance, when they are getting close, very easy recognize them as one target. We design this experiment to test the new algorithm's performance in this situation.

From Figure 2 and Figure 3, it can be seen that the tracking windows are not affected by the darker areas in upper right corner when the tracked people in the right moving upwards, so tracking is not failed. From Figure 4, it can be seen that the tracking windows can separate their respective tracking target when the two color-similar vehicles overlapped.

From the experiment results can be seen, this system can realize the real-time multi-targets tracking, and because of the application of advanced algorithm, system can effective tracking multi-targets, even at the situation that targets have similar appearance, overlapped, or the environment is complex.

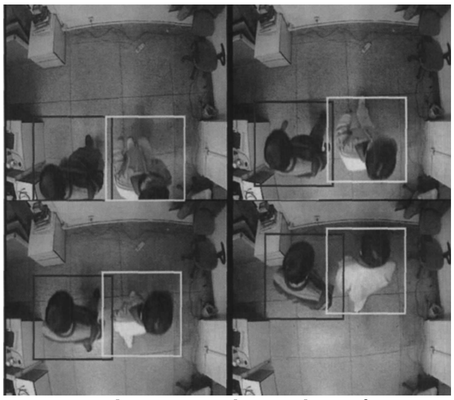

Figure 2.Multi-target Tracking with Interference Region

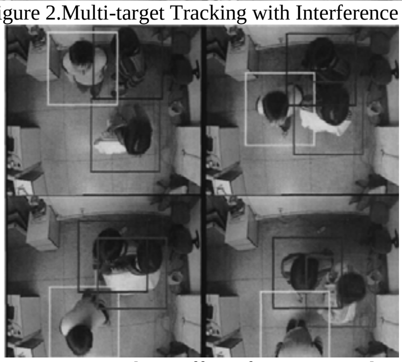

Figure 3.Tracking Effect of Targets Overlapped

Figure 4.Color-similar vehicles track when partial occlusion occurs

VII. Conclusion

Target tracking is usually non-linear and non-Gaussian, and the targets usually do voluntary movement, so their movement can not be accurately described in mathematical equations. The paper introduces the particle filter such practical estimation problem-solving method into the field of vision tracking, constructing the tracking framework based on particle filter and combining with characteristics of targets at all levels, to manufacture trackers of good performance that have "multimodal" tracking features and be able to improve robustness.

In specific implementation, re-sampling in MCMC method will be applied to solve particle degeneracy and sample impoverishment in particle filtering visual multi-target tracking algorithm. MCMC method can effetely enhance the performance of particle filter algorithm and reduce the computational complexity.

Acknowledgment

This research was supported by the Natural Science Foundation of Hebei Province under Grant No. F2012208004. The authors would like to thank the anonymous reviewers for their valuable remarks and comments.

REFERENCES

[1] Taek Lyul Song, Dong Gwan Lee, Jonha Ryu, "A probabilistic nearest neighbor filter algorithm for tracking in a clutter environment", Signal Processing, vol. 85, issue 10, pp. 2044-2053,October 2005.

[2] Mourad Oussalah, Joris De Schutter, "Hybrid fuzzy probabilistic data association filter and joint probabilistic data association filter ",Information Sciences, vol. 142, issues 1-4, pp. 195-226,May 2002.

[3] Alexandre Herbland, Raphael Lasserre, Philippe Lemetayer, Jacques Clementy, Philippe Gosse, "Malignant hypertension (MHT) : a systematic therapeutic approach through blockade of renin angiotensin system" American Journal of Hypertension, vol. 17, issue 5, pp. S155, May 2004.

[4] Arnaud Doucet, Nando De Freitas, Sequential Monte Carlo Methods in Practice. The Springer press, 2001.1-5

[5] K.Nummiaro,E.Koller-Meier,L.Van Gool, "An adaptive color-based particle filier", Image and Vision Computing, vol. 21, issue 1, pp. 99-110, 2003.

[6] Xinguang Shao, Biao Huang, Jong Min Lee, "Constrained Bayesian state estimation – A comparative study and a new particle filter based approach" , Journal of Process Control, vol. 20, issue 2, pp. 143-157,February 2010.

[7] Milstein A, Sanchez J N, Williamson E T, "Robust global localization using clustered particle filtering", Proeeedings of the National Conference on Artificial Intelligence. Edmonton, pp. 581-586, 2002.

[8] Doucet A, Gordon N J, Krishnamurthy V, "Particle filters for state estimation of jump Markov linear systems" , IEEE Transactions on Signal Processing , vol.49, pp.613-624,2001.

[9] R.T.Collins, "Mean shift Blob Tracking through Scale Space", Proc.of the IEEE CS Conference on Computer Vision and Pattern Recognition, pp. 234-240, 2003.

[10] Li Shidan, Liguo Sun, Xin Li, *Desheng Wang.* Research on an Improved Terrain Aided Positioning Model. Journal of Software, Vol 6, No 5 (2011), 937-943, May 2011.

[11] Kami Makki, Bo Sun, Ricky Guidry, Jeffery Hill. Efficient Monitoring Strategy for Active Environments. Journal of Software, Vol 6, No 4 (2011), 536-543, Apr 2011.

[12] Liubai Li. Efficient Fast Object-Tracking Scheme Based on Motion-vector-located Pattern Match. Journal of Software, Vol 7, No 5 (2012), 998-1005, May 2012.

Junying Meng Hebei Province, China. Birthdate: January, 1974. is Computer application technology M.S, graduated from Institute of Information Science&Engineerning, Hebei University of Science and Technology. And research interests on Pattern recognition and Image processing.

He is currently pursuing his Ph.D at College of Information Science and Engineering, Yanshan University.

Jiaomin Liu Henan Province, China. Birthdate: November, 1958. He received his Ph.D. from the Department of Computer Science, Hebei University of Technology, China, in 1998. Currently, he is a professor at College of Information Science and Engineering, Yanshan University, China.

His major research interests areimage processing, multimedia technology.