Journal of Theoretical and Applied Information Technology
20th April 2013. Vol. 50 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

NEURAL-BAYESIAN FILTERING BASED ON MONTE CARLO RESAMPLING FOR VISUAL ROBUST TRACKING

[1]XUNGAO ZHONG, [2*]XIAFU PENG, [3]XUNYU ZHONG

[1] PhD Student, Department of Automation, Xiamen University, Xiamen, China

[2]Prof., Department of Automation, Xiamen University, Xiamen, China

[3]Asstt Prof., Department of Automation, Xiamen University, Xiamen, China

E-mail: [1]zhongxungao@163.com, [2]pengxiafu@126.com, [3]zhongxunyu@xmu.edu.cn

*Corresponding author

ABSTRACT

The visual robust tracking is an acid test for existing methods since the target with large-dynamic-change scenarios. Specifically, this paper presents neural aided Bayesian filtering scheme which is based on Monte Carlo resampling techniques associated with lower particles hypothesis to addresses the computational intensity that is intrinsic to all particle filter (PF) approaches, including those that have been modified to overcome the degeneracy of particles and improve the diversity of particle samples. Performance and tracking quality results for severe-dynamic target tracking experiments demonstrate that the Bayesian filtering deteriorated error caused by constrain the number of particles required which was compensated by a RBF neural network, with high accuracy and intensive tracking performance for unconstrained abrupt motion only require lower particles compare with SIR Bayesian filtering, meanwhile, the proposed method is also with strong robustness for different number of particles.

Keywords: *RBF Aided Bayesian Filtering, Monte Carlo Resampling, Lower Particles Hypothesis, Visual Robust Tracking*

1. INTRODUCTION

Visual tracking is an important research issue in computer vision and have received extensive applications. In terms of tracking algorithms, the tracking processing mostly framed as a statistics estimation problem with Bayesian framework, specially, the state-estimation-based Kalman filtering (KF) [1], and sampling-based Bayesian filtering namely particle filtering (PF) [2-5].

For the scenarios with severe-dynamic motion, the KF was widely replaced with sampling-based tracking method such as the PF, for as it is a simulation-based sequential Monte Carlo sampling (SMCS) technique which capable of estimating the evolution of nonlinear, non-Gaussian stochastic processes, in addition, PF is a multiple-hypothesis solution able to estimate arbitrary distributions through evaluation of random samples in a state space. While the traditional SMCS methods face the degeneracy of particles problem, which sometimes is very severe leads to only a few particles having significant weights to represent the corresponding probability distribution.

Many extensions to PF mainly focus on the sampling methods to overcome the degeneracy of particles problems, including in [6] the importance sampling (IS)was popular replaced by Markov chain Monte Carlo (MCMC) to sample from the posterior distribution, and the adaptive MCMC [7] have shown more superiority in increasing the mixing and acceptance rates, in [8] the authors proposed a intensively adaptive MCMC (IA-MCMC) sampler to improve the sampling efficiency, which combines a density-grid-based predictive model with the stochastic approximation Monte Carlo (SAMC) algorithm [9].

On the other hand, the artificial neural network (ANN) combined with the typical PF algorithm each other have catch much attention [10], yet there is relatively little literature available describing the ANN improve PF have obtained some significant results. In [11] , during the sampling processing the particles with very big weights was splited into two small ones and use the strategy of ANN to adjust positions of tail particles in order to increase their weights，this method can adjust the distribution of particles into the area with high weight of probability distribution and raising diversity of particle samples over time. In [12], the general

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

regression neural network (GRNN) was used to adjust the samples then achieved optimize the choice of importance density.

However, some other practical problems will unavoidable appears, i.e., the primary drawback of PF and modified PF approaches are inevitable with the high computational cost associated with simulating particle and weighing multiple hypotheses [3] [13], and difficult to guarantee its robustness to dynamic noises.

In this paper, a method with Radial basis function neural network (RBFNN) aided Bayesian filtering algorithm is proposes which aims to reduce the computational cost and improves the robustness for dynamic tracking. In view of large abrupt motion tracking issues, we introduce Resampling-based Bayesian filtering, which the sampler able to estimate arbitrary distributions through evaluation of random samples in a state space. Since a certain number of samples are still required to capture the abrupt motion due to the broadness of the whole state space, which stack in favour of computational cost, therefore we further proposed a neural-Bayesian filtering based on lower sampling hypothesis, utilizes the Monte Carlo rsampling algorithm which with constraint count of particles to similar computation of the complex integration conjugate in Bayesian filtering, and the RBFNN via by compensate the Bayesian filtering error to be overcome the high computational burden caused by large number of particles problem, meanwhile the random abrupt motion cause the model unfitable which will directly decrease the tracking performance also be improved by NN. Many compare experiments demonstrated that the neural-Bayesian resampling filtering can be effective and precise tracking largely unconstrained abrupt motion target with robust for different condition with different process noises and measurement noises even using less number of sampling particles.

2. TARGET TRACKING WITH RESAMPLING-BASED BAYESIAN FILTERING

2.1 Bayesian state estimation framework

For a target tracking problem, one can consider a dynamics system described by equations, as follows

$$\begin{cases} \mathbf{X}_{(t)} = f_{(t/t\text{-}1)}\{\mathbf{X}_{(t\text{-}1)}, \varpi_{(t)}\} \\ \mathbf{Y}_{(t)} = g_{(t)}\{\mathbf{X}_{(t)}, \boldsymbol{\upsilon}_{(t)}\} \end{cases} \quad (1)$$

Where $\mathbf{X}_{(t)} \in \mathbf{R}^{n\times n}$ is the state vector, and the vector $\varpi_{(t)} \in \mathbf{R}^{n}$ is an additive disturbance affecting for dynamics system, $\mathbf{Y}_{(t)} \in \mathbf{R}^{m\times n}$ is the observation vector and $\boldsymbol{\upsilon}_{(t)} \in \mathbf{R}^{m}$ is a measurement noise vector.

Visual tracking technique generally was represented as a stochastic state estimation problem in Markov field, i.e., the dynamics equations (1) can be formulated by a Markov model with hidden state variables, therefore, the state-shift rule $f_{(t/t\text{-}1)}$ will described by a prior probability density function (PDF) $f_{(t/t\text{-}1)} \propto p(\mathbf{x}_{(t)}/\mathbf{x}_{(t\text{-}1)})$,and the state measurement function $h(t)$ is described by a likelihood PDF $g_{(t)} \propto p(\mathbf{x}_{(t)}/\mathbf{Y}_{(t)})$ .If obtained a series of independent observations information $\mathbf{Y}_{(1:t)}$ up to time t, which is represented by a posterior PDF $\mathbf{Z}_{(1:t)} \propto p\{\mathbf{X}_{(t)}/\mathbf{Y}_{(1:t)}\}$, according to the Bayesian filtering framework ,the accurate estimate of the posterior PDF at each time step will constitutes a complete solution for state estimation equation given by[11]

$$\begin{cases} p\{\mathbf{X}_{(t)}/\mathbf{Y}_{(1:t\text{-}1)}\} = \int f_{(t/t\text{-}1)}\, p\{\mathbf{X}_{(t\text{-}1)}/\mathbf{Y}_{(1:t\text{-}1)}\} \mathrm{d}\mathbf{X}_{(t\text{-}1)} \\ p\{\mathbf{X}_{(t)}/\mathbf{Y}_{(1:t)}\} = \dfrac{g_{(t)}\{\mathbf{Y}_{(t)}/\mathbf{X}_{(t)}\}\, p\{\mathbf{X}_{(t)}/\mathbf{Y}_{(1:t\text{-}1)}\}}{p\{\mathbf{Y}_{(t)}/\mathbf{Y}_{(1:t\text{-}1)}\}} \end{cases} \quad (2)$$

Where

$$p\{\mathbf{Y}_{(t)}/\mathbf{Y}_{(1:t\text{-}1)}\} = \int g_{(t)}\{\mathbf{Y}_{(t)}/\mathbf{X}_{(t)}\}\, p\{\mathbf{X}_{(t)}/\mathbf{Y}_{(1:t\text{-}1)}\} \mathrm{d}\mathbf{X}_{(t)} \quad (3)$$

Then the minimum-variance state estimation for dynamics system, as follows

$$\hat{\mathbf{X}}_{(t)} = \int \mathbf{X}_{(t)}\, p\{\mathbf{X}_{(t)}/\mathbf{Y}_{(1:t)}\} \mathrm{d}\mathbf{X}_{(t)} \quad (4)$$

2.2. Resampling-based Bayesian filtering for posterior PDF estimation

Since the complex integration computation associate with Bayesian filtering equations(2)-(4) are none analytical solution in practice, fortunately, sampling-based technique will engenders the application of Bayesian filtering possibility, such as famous Monte Carlo sequential importance resampling (SIR) [14] that is recursively estimates posterior PDF $p\{\mathbf{X}_{(t)}/\mathbf{Y}_{(1:t)}\}$ by selecting and simulating a statistically relevant subset of possible

Journal of Theoretical and Applied Information Technology
20th April 2013. Vol. 50 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

system states, formally, the goal of SIR is to obtain a set of **N** discrete sample $\{\tilde{\mathbf{X}}^{i}_{(t)}\}_{i=1}^{N}$ and their corresponding weights $\{\tilde{\mathbf{W}}^{i}_{(t)}\}_{i=1}^{N}$ to approximate posterior PDF at time *t*, then the integration in Bayesian filtering equation (2) can be approximated by point masses, as follows[6]

$$p\{\mathbf{X}_{(t)}/\mathbf{Y}_{(1:t)}\} \approx \sum_{i=1}^{N} \tilde{\mathbf{W}}^{i}_{(t)}\delta(\mathbf{X}_{(t)} - \mathbf{X}^{i}_{(t)}) \tag{5}$$

The particles-weights pair can then be used to compute an estimate of the system state $\hat{\mathbf{X}}_{(t)}$, given by

$$\hat{\mathbf{X}}_{(t)} \approx \sum_{i=1}^{N} \tilde{\mathbf{W}}^{i}_{(t)}\tilde{\mathbf{X}}^{i}_{(t)} \tag{6}$$

As shown in algorithm 1, we introduce a sequential importance resampling Bayesian filtering (SIRBF), which an implementation of the Markov china Monte Carlo resampling algorithm, the resampling-based tracking method aims to enlarge the sampling variance to cover the possible motion uncertainty will be a direct solution for the visual dynamic tracking.

Algorithm 1. SIRBF for Visual Tracking

// sampling:
For *i*=1:N do
Particles sampling:

$$\tilde{x}^{i}_{(t)} \sim q(\tilde{x}^{i}_{(t)}/\tilde{x}^{i}_{(t-1)}, y_{(t)}), \{(\tilde{x}^{i}_{(t)}, \tilde{w}^{i}_{(t)})\}_{i=1}^{N}$$

// weights updating:
For *i*=1:N do
Update the working weights:

$$\tilde{w}^{i}_{(t)} = \tilde{w}^{i}_{(t-1)}\frac{p(y_t/\tilde{x}^{i}_{(t)})\,p(\tilde{x}^{i}_{(t)}/\tilde{x}^{i}_{(t-1)})}{q(\tilde{x}^{i}_{(t)}/\tilde{x}^{i}_{(t-1)}, y_{(t)})}$$

Normalize the weights:

$$\tilde{w}^{i}_{(t)} = \tilde{w}^{i}_{(t)} / \sum_{j=1}^{N}\tilde{w}^{j}_{(t)}$$

// Resampling:

Let $N_{eff} = 1/\sum_{j=1}^{N}(\tilde{w}^{j}_{(t)})^2$

For *i*=1:N do
If N_{eff} < threshold
Resampling particle set:

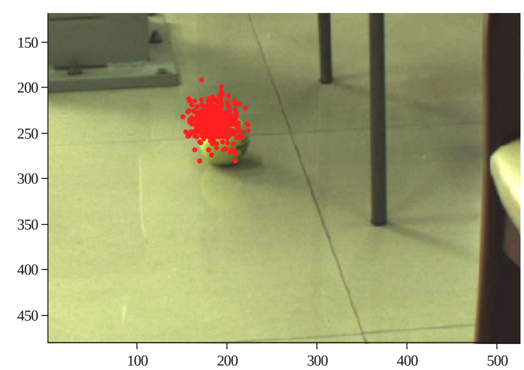
$$\{(x^{i}_{(t-1)}, w^{i}_{(t-1)})\}_{i=1}^{N} = \{(x^{i}_{(t-1)}, 1/N)\}_{i=1}^{N}$$

Else $\{(x^{i}_{(t-1)}, w^{i}_{(t-1)})\}_{i=1}^{N} = \{(\tilde{x}^{i}_{(t)}, \tilde{w}^{i}_{(t)})\}_{i=1}^{N}$

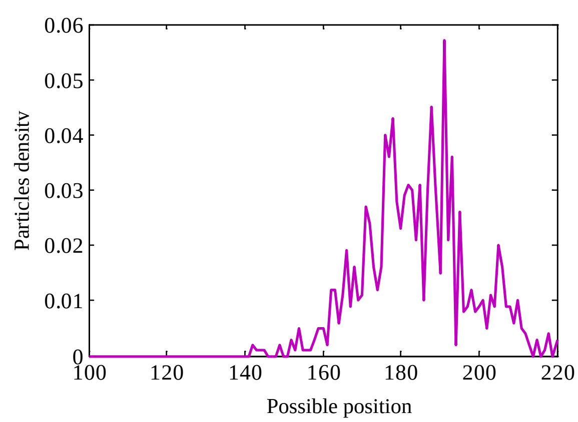
// position estimation:
Target tracking according to equation (6)

The SIRBF result shown in Fig.1, the possible target position will be presented by sampling particles with their corresponding weights, i.e., the success of the resampling-based Bayesian filtering highly relies on its ability to maintain a good approximation to the posterior distribution, but there exist potential primary drawback of sampling approaches for a large number of particles are required to guarantee sufficient sampling in the broad state space. Meanwhile, as shown in Fig 2, the tracking accuracy will improves with the count of particles increase.

Therefore the high computational burden caused by a large number of particles often makes the resampling-based Bayesian filtering infeasible for practical applications. In view of above problems, we proposed a method associated with neural networks to aids the resampling-based Bayesian filtering to reduce the particle count, while maintaining tracking quality and the computational demands remain lower compare with the tradition particle filtering.



(a)



Journal of Theoretical and Applied Information Technology
20th April 2013. Vol. 50 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

(b)

Fig.1. Illustration Of The Possible Target Position Was Presented By Particles With Number Of 500, (a) Result Of SIR, (b) The Density Distribution Of Sampling Particles Meet Normal Distribution.

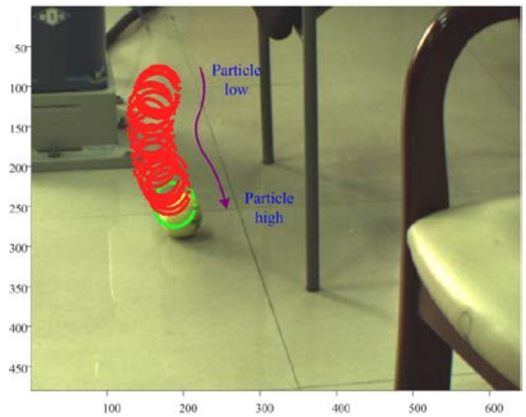


Fig.2. The Estimation Accuracy Of Possible Target Position Against To The Different Number Of Particles (From 50 To 1500) With SIR Algorithm.

3. ARTIFICIAL NEURAL NETWORK AIDED BAYESIAN FITERING

Bayesian filtering tracking techniques involve the simulation of multiple hypotheses, real-time performance remains challenge. The existing solutions in [15] aims to lower the dimensionality of the problem; in [16] have been modified to minimize the number of particles meeting to reduce the computational cost.

Constrained the count of particles with lower hypotheses is the effective resolution for lower the computational cost for Bayesian filtering. Here we present a methodology using constrained particles-weights pair to approximate posterior PDF and the deteriorated performance causes by Minimized particles was compensated via aided by a ANN.

The posterior PDF $p\{\mathbf{X}_{(t)}/\mathbf{Y}_{(1:t)}\}$ of the target position will be optimal approximated by selecting and simulating statistically relevant subsets with enough larger numbers, however, number-constrained particle subsets will directly leading to deteriorated errors caused by absent particles. On the Other hand, in actual tracking process, the covariance $\mathbf{Q}_{(k)}$ and $\mathbf{R}_{(k)}$ of process noises $\varpi_{(t)}$ and visual measurement $\upsilon_{(t)}$ are difficult to be definitely pinpointed. There are parameter uncertainty errors during Bayesian filtering processing. Therefore, the desired posterior PDF $p^*\{\mathbf{X}_{(t)}/\mathbf{Y}_{(1:t)}\}$ of the target position should be given by

$$p^*\{\mathbf{X}_{(t)}/\mathbf{Y}_{(1:t)}\} = \hat{p}\{\mathbf{X}_{(t)}/\mathbf{Y}_{(1:t)}\} + \Delta p'\{\mathbf{X}_{(t)}/\mathbf{Y}_{(1:t)}\} + \Delta p''\{\mathbf{X}_{(t)}/\mathbf{Y}_{(1:t)}\} \quad (7)$$

Where the errors $\Delta p'\{\mathbf{X}_{(t)}/\mathbf{Y}_{(1:t)}\}$ and $\Delta p''\{\mathbf{X}_{(t)}/\mathbf{Y}_{(1:t)}\}$ are refer to deteriorated error causes by Minimized particles and approximated noises' statistics, respectively.

In this paper a method to compensate for those errors is proposed to improve the estimation accuracy of the Bayesian filtering with lower particles by incorporating the neural network into the state estimation stage. The structure of the neural-Bayesian filtering based on SIR appears in Fig. 3. To simplify notations, we named this tracker as neural-Bayesian resampling filtering (NBRF).

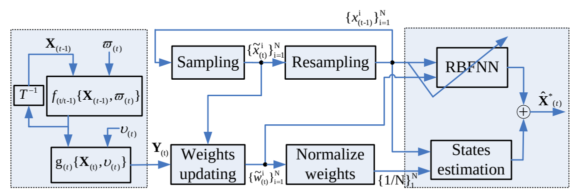


Fig.3. the Structure of NBRF

The neural network we chosen with one hidden layer which is the most widely spread architecture type, and the activation function of the hidden nodes is chosen to be a radial basis function, the output of each hidden neuron with Gaussian basis function is defined as

$$n_i^1 = \exp(-\left\|\mathbf{W}^1 - \mathbf{G}\right\|^2 \bullet b_i^1) \quad (8)$$

$$\left\|\mathbf{W}^1 - \mathbf{G}\right\| = [(\mathbf{W}^1 - \mathbf{G}^{\mathbf{T}})(\mathbf{W}^1 - \mathbf{G}^{\mathbf{T}})^{\mathbf{T}}]^{1/2} = \left(\sum_{i=1}^{k}(w_i^1 - g_i)^2\right)^{1/2} \quad (9)$$

Where $\mathbf{G} \in R^{m\times n}$ is the input samples set and $\mathbf{W}^1 \in R^{n\times m}$ is the neuron weight, b_i^1 is the threshold of *i*th neuron in the hidden layer? The output of the network is the linear sum of the outputs of the hidden neurons. So, the compensation for the errors for target position estimation is approximated as

$$\Delta\hat{\mathbf{P}}^*_{(t)} = \sum_{j=1}^{n} w_j^2 n_j^1 + b_i^2 \quad (10)$$

Where w_i^2 and b^2 are the weights and threshold of output layer, respectively.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The learning algorithm will be applied to determine the structure of neural network, and the learn goal is to obtain the convergence of the thresholds **b** and the connection weight **W** of the input vector to hidden layer and the hidden layer to output layer and for each layer. We are choose a radial basis function neural network (RBFNN) whose training is a partial approximate process, and less time is cost during its training procedure in comparison of the BP network. The learning law of the network is given by

$$w_{ij}^{l}(k) = w_{ij}^{l}(k-1) - \alpha \frac{\partial E}{\partial w_{ij}^{l}(\mathrm{k}-1)} \tag{11}$$

$$E = \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{M}(y_{nm} - O_{nm})^2 \tag{12}$$

Where y_{nm} is the output of the m^{th} neurons of output layer with n^{th} input sample, O_{nm} is the desired output of output layer, α is the learning rate, and the root mean square error(RMSE) is choose as iterative termination conditions, given by

$$RMSE = \left(\sum_{l=1}^{N}\sum_{k=1}^{Q}(O_{lk} - y_{lk})^2 / \mathrm{QN}\right)^{1/2} \tag{13}$$

The neural network learning processing was given as following:

Step1).Initialized the thresholds and weights.

Step2).calculate the output and the root mean square error according to equation (12), (13), respectively.

Step3).if $RMSE \leq \zeta$, go to step 5), else next step.

Step4).adjust the thresholds and weights according to equation (11), go to step 2).

Step5).Saving the thresholds and weights in an array iterative end.

4. EXPERIMENT RESULTS AND DISCUSSUIN

To test the empirical performance of our tracking approach, we collected several motion sequences that involve severe dynamic in various scenarios, which including sudden dynamic changes, and the changes with downsampling videos. All the experiments we implemented to compare the tracking performance of our tracker, i.e., NBRF with SIRBF are with different count of particles. For the sake of fair comparison, the important function we choose is the same Gaussian kernels $N\left(\mathbf{R}, (\mathbf{Y}_{(t)} - \hat{\mathbf{X}}_{(t)})^2\right)$.

To qualitatively evaluate the tracking performance of NBRF, here we are shown our experiments are a bounced table ball that back and force struck the floor with sudden dynamic changes. The unexpected motion dynamic makes the tracking task rather hard by an accurate motion model. Our experiments to illustrate the proposed NBRF approach can effectively deal with this difficulty only using lower particles which aided by RBFNN.

Sample frames are shown in Fig.4, with 100 particles, our method NBRF (Fig.4 (c)) successfully tracked the bouncing pingpong throughout the sequence. Note that even with 1000 samples, SIRBF (Fig. 4(b)) poorly performs, experiencing a significant drift of the target object. Moreover, SIRBF (Fig. 4(a)) failed to track the table ball in most frames using the number of samples lower 150 due to the large motion uncertainty.

We then perform a quantitative comparison of tracking accuracy between SIRBF and NBRF with different particles to further verify that the use of proposal in FBFNN does help on a similarly test video. The comparison is based on the position error in pixels, as follows

$$e_i = \left\|(x_p^i, y_p^i) - (x_g^i, y_g^i)\right\|, i = 1, 2, 3\ldots \tag{14}$$

Where (x_p^i, y_p^i) is the estimation value of target position by SIRBF or NBRF, (x_g^i, y_g^i) is the ground-truth position. The frame-by-frame comparison of the position error in pixels for those two trackers is shown in Fig.5. It can be seen that compared with the SIRBF tracking result, our method NBRF is more closer to the ground-truth position, it means that with the aid of RBFNN the performance of NBRF is better than the stand-alone SIRBF. The experiments shown the neural network we chose to compensate for the error of SIRBF, although even with lower particles, the NBRF always work well. This means the RBFNN have improved the robustness of SIRBF for severe dynamic changes.

Another experiment is to qualitatively evaluate the tracking performance of the NBRF and SIRBF on a synthetic sequence that involves the severe abrupt motion of the object caused by frames inconsecutive and low-frame-rate video. In this experiment, the same 100 samples are used for

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

NBRF and SIRBF. Sample frames are illustrated in Fig.6. It is observed that our approach can effectively track the object throughout the sequence even the ball bounced back with sudden dynamic changes.

On the other hand, SIRBF frequently lose the track and poorly perform on this sequence due to the large motion uncertainty. It means that the performance of the NBRF is better than the SIRBF, and it also illustrate the neural network plays an important role in error compensation to improve the

SIRBF tracking ability, experiments proved that our proposed NBRF with intensive tracking performance for largely unconstrained abrupt motion even with less number of particles.

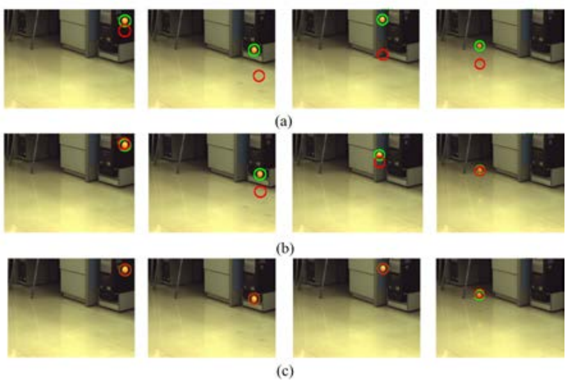


Fig.4. Tracking Performances Of The two Trackers On Dynamic Changes Video(Green Target Position, Red Tracking Result),(a) SIRBF With 150 Particles,(b)SIRBF With 1000 Particles, (c) NBRF With 150 Particles.

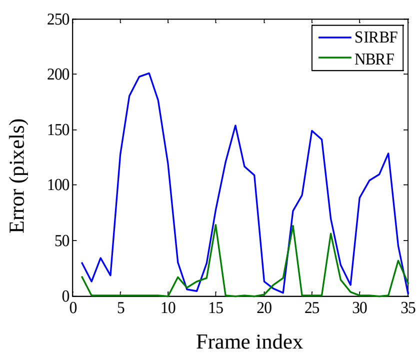


Fig.5. Corresponding Of Position Error In Pixels For The SIRBF And NBRF With 100 Particles.

For the sake of test our proposed tracker's robustness for different number of particle, many other tracking experiments with the video same as in Fig.6, the successful tracking rate with two tracker compare in Table 1, which demonstrated that the tracking performance of SIRBF is sensitive to the number of particles, it is worth noting that if particles less 50 the SIRBF will lost the tracking ability, and with more than 1000 particles the successful tracking rate will towards stability 83%. While the NBRF only with lower 100 particles performed good tracking which the same as SIRBF with 1000 particles. SO, we can kindly gets conclusion that our method is a robust tracker no matter with lower particles could be intensive tracking the unconstraint motion.

Table 1. The Successful Tracking Rate (×100%) Of SIRBF And NBRF With The Different Particles On Sudden Dynamic Changes With Low-Frame-Rate Video

Method	The number of particles				
	100	300	500	1000	>1000
SIRBF	0.41	0.57	0.65	0.81	0.83
NBRF	0.87	0.91	0.95	0.96	0.96

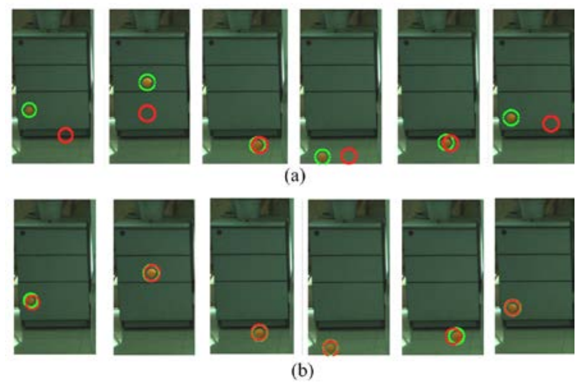


Fig. 6. Compare The Tracking Results Of SIRBF And NBRF With The Smae100 Particles On The Sudden Dynamic Changes With Low-Frame-Rate Video (Green Target Position , Red Tracking Result).

5. CONCLUSION

In this paper, a method RBFNN aided sequential importance resampling Bayesian filtering (SIRBF) has presented to improve the performance for robust dynamic tracking with lower particles hypothesis. In the proposed neural-Bayesian framework, a novel method with neural network(NN) merge together with resampling-based Bayesian filtering, and the NN has useful to improve the tracking precise even with less particles. Many compare experiments illustrate that the neural-Bayesian resampling filtering (NBRF) with robust tracking performance for largely unconstrained abrupt motion only require lower number of particles compare with SIRBF, on the other hand, our method also with strong adaptive for different number of particle and dynamic noises.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ACKNOWLEDGMENTS

This work was supported in part by the Fujian Provincial Natural Science Foundation of China (NO.2010J05141).

REFRENCES:

[1] R. E. Kalman, "A new approach to linear filtering and prediction problems", Trans. ASME, Ser. D. J. Basic Eng., Vol.82, 1960,pp. 34-45.

[2] G. Gualdi, A. Prati, and R. Cucchiara, "Multistage Particle Windows for Fast and Accurate Object Detection", *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.34, No.8, 2012, pp.1589-1604.

[3] J. Anthony Brown and David W. Capson, "A framework for 3D model-based visual tracking using a GPU-accelerated particle filter", *EEE Trans. Visualization and Computer Draphics*, Vol.18,No.1,2012,pp.68-80.

[4] L. Kratz, and K. Nishion,"Tracking pedestrians using local spatiotemporal motion patterns in extremely crowded scences", *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.34, No.5, 2012, pp.987-1002.

[5] L. Bai, Y. Wang , "Road tracking using particle filters with partition sampling and auxiliary variables", *Computer Vision and Image Understanding*, Vol.115,No.10,2011, pp. 1463–1471.

[6] Z. Khan, T. Balch, and F. Dellaert, "An MCMC-based particle filter for tracking multiple interacting targets", *in Proc. ECCV*, 2004,pp.279-290.

[7] Y.Atchad and J. S.Rosenthal, "On adaptive Markov chain Monte Carlo algorithms", *Bernoulli*, Vol.11, No.5, 2005, pp. 815-828.

[8] X. Zhou, Y. Lu, J. Lu, J. Zhou, "Abrupt Motion Tracking Via Intensively Adaptive Markov-Chain Monte Carlo Sampling", *IEEE Trans. image processing*, Vol.21,No.2,2012,pp.789-801.

[9] F. Liang, C. Liu, and R. J. Carroll, "Stochastic approximation in Monte Carlo computation", *J. Amer. Statist. Assoc.*, Vol.102, 2007, pp. 305-320.

[10] T. Ogura, C. Sato, "Automatic particle pickup method using a neural network has high accuracy by applying an initial weight derived from eigenimages: a new reference free method for single-particle analysis", *J. Structural Biology*, vol.145, No.1,2004, pp. 63-75.

[11] Zhonyu P., Derong L., Ning J., and Zhuo W., "A Monte Carlo particle model associated with neural networks for tracking problem", *IEEE Trans Circuits and Systems-I: Regular papers*, Vol.55,No.11,2008,pp. 3421-3429.

[12] Qin, W. ,Peng, Q C , "An improved particle filter algorithm based on neural network for visual tracking", *International Conference on Communications, Circuits and Systems Proceedings*,2007, pp.765-768.

[13] A.C. Sankaranarayanan, A. Srivastava, and R. Chellappa. "Algorithmic and architectural optimizations for computationally efficient particle filtering", *IEEE Trans. Image Processing*, Vol.17, No.5, 2008, pp. 737-748.

[14] M. Isard and A. Blake, "Condensation conditional density propagation for visual tracking", *Int. J. Comput. Vis.*, Vol.29, No.1, 1998, pp. 5-28.

[15] Y. Wu, J. Lin, and T.Huang, "Analyzing and Capturing Articulated Hand Motion in Image Sequences", *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.27,No.12, 2005,pp. 1910-1922.

[16] M. Bray, E. Koller-Meier, and L. Van Gool, "Smart particle filtering for high-dimensional tracking", *Computer Vision and Image Understanding*, Vol.106, No.1, 2007, pp. 116-129.