TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol.12, No.6, June 2014, pp. 4731 ~ 4739
DOI: 10.11591/telkomnika.v12i6.5493

Severe-Dynamic Tracking Problems Based on Lower Particles Resampling

Xungao Zhong[1], Xiafu Peng*[2], Xunyu Zhong[3]
Department of Automation, Xiamen University
Siming Road, Xiamen 361005, China, Tel.: +86-137-2088-5429, Fax: +86-592-258-0005
*Corresponding author, e-mail : zhongxungao@163.com[1], pengxiafu@126.com[2], zhongxunyu@xmu.edu.cn[3]

Abstract

For a target as it with large-dynamic-change which is still challenging for existing methods to perform robust tracking; the sampling-based Bayesian filtering often suffer from computational complexity associated with large number of particle demanded and weighing multiple hypotheses. Specifically, this work proposes a neural auxiliary Bayesian filtering scheme based on Monte Carlo resampling techniques, which to addresses the computational intensity that is intrinsic to all particle filter, including those have been modified to overcome the degeneracy of particles. Tracking quality for severe-dynamic experiments demonstrate that the neural via compensate the Bayesian filtering error, with high accuracy and intensive tracking performance only require lower particles compare with sequential importance resampling Bayesian filtering, meanwhile, our method also with strong robustness for low number of particles.

Keywords*: Bayesian filtering, neural network, Monte Carlo resampling, particles constrained, robust tracking*

1. Introduction

Robust tracking is an active research topic in computer vision, and it has received extensive applications including in intelligent surveillance, robot visual servoing, etc. In terms of tracking algorithms, large number of approaches have been proposed in the past decades, such as the state-estimation-based Kalman filtering (KF) [1], and sampling-based Bayesian filtering, namely particle filtering (PF) [2-4].

For the scenarios with severe-dynamic motion, the KF was widely replaced by sampling-based tracking method such as the PF, which is a multiple-hypothesis solution able to estimate arbitrary distributions through evaluation of random samples in a state space, therefore, sampling is a vital step for PF, while the traditional sequential Monte Carlo sampling(SMCS) methods face the degeneracy of particles problem, which sometimes is very severe leads to only a few particles are used to represent the corresponding probability distribution.

Therefore, the extensions to PF, recently, mainly focus on the sampling methods to overcome the degeneracy of particles problems and to improve the diversity of particle samples, including auxiliary variable PF (AVPF) [5], fission bootstrap PF (FBPF)[6]. In order to further improve the sampling efficiency, the famous Markov chain Monte Carlo (MCMC) method have obtained the considerable development [7, 8], and the adaptive MCMC [9, 10] have shown more superiority in increasing the mixing and acceptance rates, in [11] 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 [12].

In this paper, a method with radial basis function neural network (RBFNN) auxiliary particle filtering algorithm is proposes, which was motivated by reduces the computational cost and improves the robustness for visual tracking, and we apply this neural-particle filtering schema in large abrupt motion tracking problems. We first having a description to the Bayesian state estimation framework for visual tracking task, then we introduce sampling-based Bayesian filtering, which the sampler able to estimate arbitrary distributions through evaluation of random samples in a state space. But for the unconstrained abrupt motion tracking problem, a certain

Received December 29, 2013; Revised March 7, 2014; Accepted March 22, 2014

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 resampling filtering (NBRF) based on lower sampling hypothesis, utilizes the Monte Carlo sampling algorithm which with constraint count of particles to similar computation of the complex integration conjugate in Bayesian filtering, and the neural network (NN) via by compensate the Bayesian filtering error can 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 NBRF can be effective and precise tracking largely unconstrained abrupt motion with robust even using less number of sampling particles.

2. Resampling-Based Bayesian Filtering for Tracking Problem

For a target as it with abrupt motion, it is a challenging problem to achieve the robust tracking, since the severe dynamic target was difficult represented by a linear approximated-model. Herein, we introduce the resampling-based Bayesian filtering methods which is to enlarge the sampling variance to cover the possible motion uncertainty.

Sampling technique such as Monte Carlo sequential importance resampling (SIR) [13] that is recursively estimates posterior probability density function (PDF) $p\{\mathbf{X}_{(t)}/\mathbf{Z}_{(1:t)}\}$ by selecting and simulating a statistically relevant subset of possible system states, formally, the goal of SIR is to obtain a set of N discrete sample $\{\tilde{\mathbf{X}}^{i}_{(t)}\}^{N}_{i=1}$ and their corresponding weights $\{\tilde{\mathbf{W}}^{i}_{(t)}\}^{N}_{i=1}$ to approximate posterior PDF, then the integration in Bayesian filtering can be approximated by point masses, as follows [12]:

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

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

$$\hat{\mathbf{X}}_{(t)} \approx \sum_{i=1}^{N} \tilde{\mathbf{W}}^{i}_{(t)} \tilde{\mathbf{X}}^{i}_{(t)} \tag{2}$$

As shown Algorithm 1, we introduce a sequential importance resampling Bayesian filtering (SIRBF), which an implementation of the Markov china Monte Carlo resampling algorithm for tracking problems.

Algorithm 1. SIRBF for target tracking

for i=1:N do // Particles sampling with SIR

$\tilde{x}^{i}_{(t)} \sim q(x^{i}_{(t)}/\tilde{x}^{i}_{(1:t-1)}, \mathrm{z}_{(1:t)})$, and set $\tilde{x}_{(1:t)} \overset{\Delta}{=} \left(x^{i}_{(1:t-1)}, \tilde{x}^{i}_{(t)}\right)$

end for

for i=1:N do // weights updating

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

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

end for

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

if Neff < Nthreshol

for i=1:N do // Resampling with SIR

$\{(x_{(t)}^{i}, w_{(t)}^{i})\}_{i=1}^{N} = \{(x_{(t)}^{i}, 1/N)\}_{i=1}^{N}$

end for

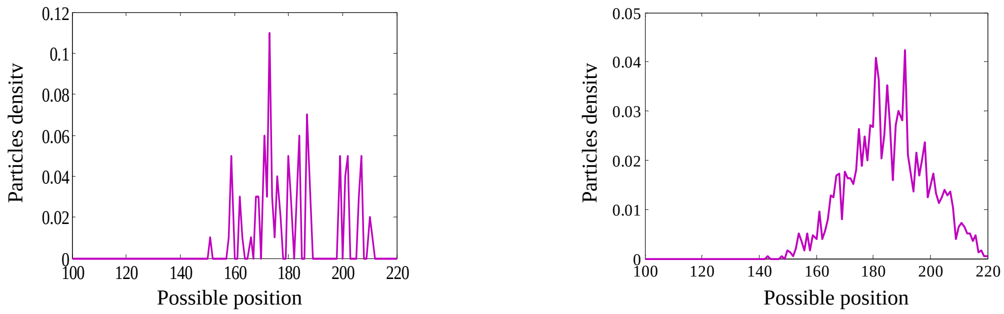
else $\{(x_{(t)}^{i}, w_{(t)}^{i})\}_{i=1}^{N} = \{(\tilde{x}_{(t)}^{i}, \tilde{w}_{(t)}^{i})\}_{i=1}^{N}$

The SIRBF results are shown in Figure 1, the possible target position is presented by sampling particles with their corresponding weights, i.e., the success of the SIRBF 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, and the estimation accuracy is linearly with the number of particles, as shown in Figure 2(a) the tracking accuracy will improves with the count of particles increase, Figure 2(b) show that the computational cost is fit with cubic polynomial based on number of sampling, given by:

$$f(x) = -4.335\times10^{-14}x^3 + 1.062\times10^{-8}x^2 + 3.947\times10^{-6}x + 6.749\times10^{-5} \quad (3)$$

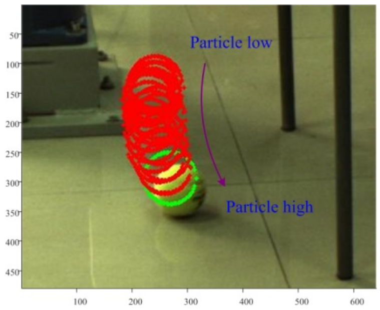
Therefore, the high computational burden caused by a large number of particles often makes the SIRBF infeasible for practical applications. In view of above problems, we proposed a method associated with neural networks to aids the resampling-based Bayesian filtering to reduce the particle count, while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

(a)

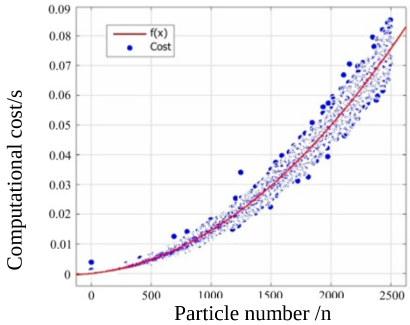


(b)

Figure 1. Illustration of the Possible Target Position was Presented by Particles with Number 100, 1500 Respectively, the Estimation Accuracy of Target Position is Linearly with the Number of Particles, (a) result of SIRBF, (b) the density distribution of sampling particles meet normal distribution



(a)



(b)

Figure 2. Illustration of the Estimation Accuracy of Possible Target Position Against to the Computational Cost of the SIRBF Algorithm, (a) the estimation result of SIR under different number of particles (from 50 to 1500), the tracking accuracy is increased with increasing particle number, (b) the computational cost is fitting with quadratic based on number of sampling, the cost fit with function $f(x)=a_3*x^3+a_2*x^2+a_1*x+a_0$.

3. Neural Network Auxiliary Particle Filtering

SIRBF algorithm involves the simulation of multiple particles hypotheses, in scene to high dimensionality of model-based tracking, the real-time performance remains challenge [14]. The solutions in [4] proposes a GPU-accelerated PF for 3D visual tracking application, in [15] aim to lower the dimensionality of the problem, and [16] have been modified to minimize the number of particles meeting to reduce the computational cost.

Constrained the counts of particles with low-level hypotheses is the effective resolution for lower the computational cost for SIRBF, however, a certain number of samples are still necessary to capture the severe dynamic motion due to the broadness of the whole state space, so, simply decrease the number of particles inevitable severe deteriorate the tracking accuracy. Herein we present a methodology using constrained particles-weights pair to approximate estimation the posterior PDF, and the deteriorated performance causes by minimized particles was compensated via aided by radial basis function neural network (RBFNN).

According to the Equation (2) 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. Therefore, the desired target position should be given by:

$$\hat{\mathbf{X}}^*(t) = \hat{\mathbf{X}}(t) + \Delta\mathbf{X}_{par}(t) \tag{4}$$

Where $\Delta\mathbf{X}_{par}(t)$ refers to deteriorated error causes by Minimized particles. In this paper a method to compensate for the errors $\Delta\mathbf{X}_{par}(t)$ is proposed to improve the estimation accuracy of the SIRBF with lower particles by incorporating the RBFNN into the state estimation stage. As appears Figure 3, the RBFNN is embed into resampling-based Bayesian filtering to overcome he high computational burden caused by a large number of particles. The algorithm of neural-Bayesian resampling filtering (NBRF) consists of system model, SIR sampling, state estimation.

The RBFNN 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:

$$n_i^1 = \exp(-\left\|\mathbf{C}^1 - \mathbf{G}\right\|^2 \bullet b_i^1) \tag{5}$$

$$\left\|\mathbf{C}^1 - \mathbf{G}\right\| = [(\mathbf{C}^1 - \mathbf{G}^{\mathbf{T}})(\mathbf{C}^1 - \mathbf{G}^{\mathbf{T}})^{\mathbf{T}}]^{1/2} = \left(\sum_{i=1}^{k}(c_i^1 - g_i)^2\right)^{1/2} \tag{6}$$

Where $\mathbf{G} \in R^{m\times n}$ is the input samples set and $\mathbf{C}^1 \in R^{n\times m}$ is the central vector with the ith element denoted as $g_i \in R^{m\times 1}$, $c_i \in R^{m\times 1}$,respectively; $b_i^1$is the threshold of ith 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:

$$\Delta \mathbf{X}_{par}(t) = \sum_{j=1}^{n} w_j^2 n_j^1 + b_i^2 \quad (7)$$

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

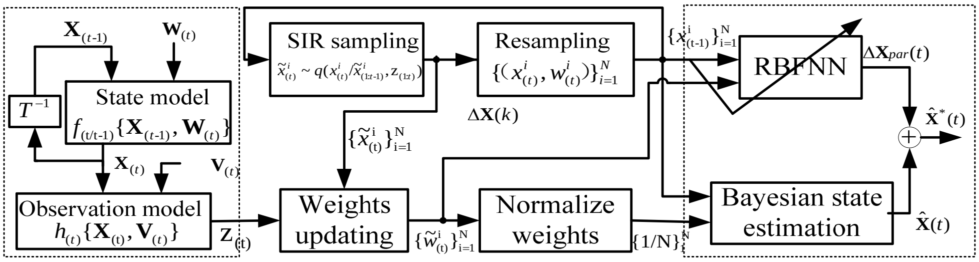


Figure 3. The Structure of Neural-Bayesian Resampling Filtering (NBRF)

4. Experimental Results

To test the empirical performance of our tracking approach, we collected several motion sequences that involve severe dynamic in various scenarios, which including the low-frame-rate, sudden dynamic changes, and the sudden dynamic changes with downsampling videos. All the experiments we implemented to compare the tracking performance of our tracker, i.e., NBRF and SIRBF with different count of particles.

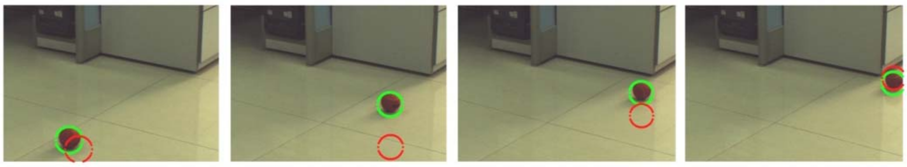
Scenario 1 low-frame-rate video: We have first investigated how the number of particle affects the tracking performance of SIRBF and how can our proposed NBRF to improve the tracking precise even with lower particles. In this experiments, the test videos with random smooth motion, which a tennis running on the floor, we begin by giving the comparison of the tracking performance of SIRPF and NBRF on a low-frame-rate video squash, which is downsampled by keeping one frame in every 10 frames from a original video. In order to qualitatively evaluating the impact of the count of particles, we test SIRBF with samples 100 and 1000 scenarios, and the sample frames of result are shown in Figure 4. In Figure 4(a), the performance of SIRBF with lower 100 particles is bad, even lost the tracking due to the abrupt motions caused by severe frame dropping, whereas when the particles increasing to 1000 the result of SIRBF (Figure 4(b)) is accurately tracking the ball after frame 19th(original frame 190), but this at the cost of large number of particles, while our NBRF method (Figure 4(c)) can effectively dealt with this difficulty using only lower 100 samples, and keeping more accurately performed than SIRBF throughout the sequence.

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

$$e_i = \left\| (x_p^i, y_p^i) - (x_g^i, y_g^i) \right\|, \ \mathrm{i} = 1,2,3\ldots \quad (8)$$

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. Tracking error is evaluated as the difference between the measurement position and the position estimated by the SIRBF or NBRF. The tracking performance of the SIRBF and the NBRF with different particles are compared in Figure 5, the position error of NBRF is apparently lower than that of SIRBF with the same particles, as shown the result with 100 particles. On the other hand, it is worth note that all five experiments demonstrate our

NBRF performed higher accuracy than SIRBF even with lower particles, and the SIRBF performed same as NBRF but cost more than 10 times number of particles. We believe that the improved tracking performance of NBRF is mainly due to the proposal RBFNN to compensate the errors during the constrain particles, to cover the possible motion uncertainty will be a direct solution to improve the tracking precise, meanwhile, the RBFNN can be well decrease the computational cost via by constrain the number of particles and keeping the well performances.

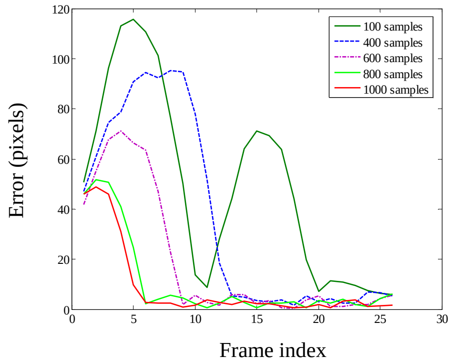


(a)

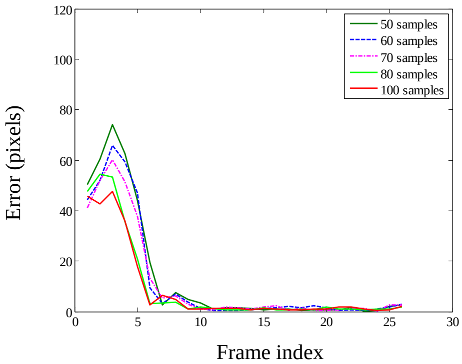
(b)

(c)

Figure 4. Tracking Performances of the Two Trackers on Low-frame-rate Video (green target position, red tracking result), (b) SIRBF with 100 particles, (c)SIRBF with 1000 particles, (d) our NBRF with 100 particle



(a)



(b)

Figure 5. Tracking Error with Different Particle Conditions, (a) SIRBF, (b) NBRF

Scenario 2 bounced ball with dynamic changes: To further qualitatively evaluate the tracking performance of our NBRF, where 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.

Sample frames are shown in Figure 6, with 100 samples, our NBRF method (Figure 6(c)) successfully tracked the bouncing pingpong throughout the sequence. Note that, even with 1000 samples, SIRBF method (Figure 6(b)) performed poor tracking ability, experiencing a significant drift of the target object. Moreover, SIRBF (Figure 6(a)) failed to track the table ball in most frames using the number of samples lower 150.

The frame-by-frame comparison of the position error in pixels for those two trackers is shown in Figure 7. 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 SIRBF, due to the RBFNN can improve the robustness of NBRF for severe dynamic changes.

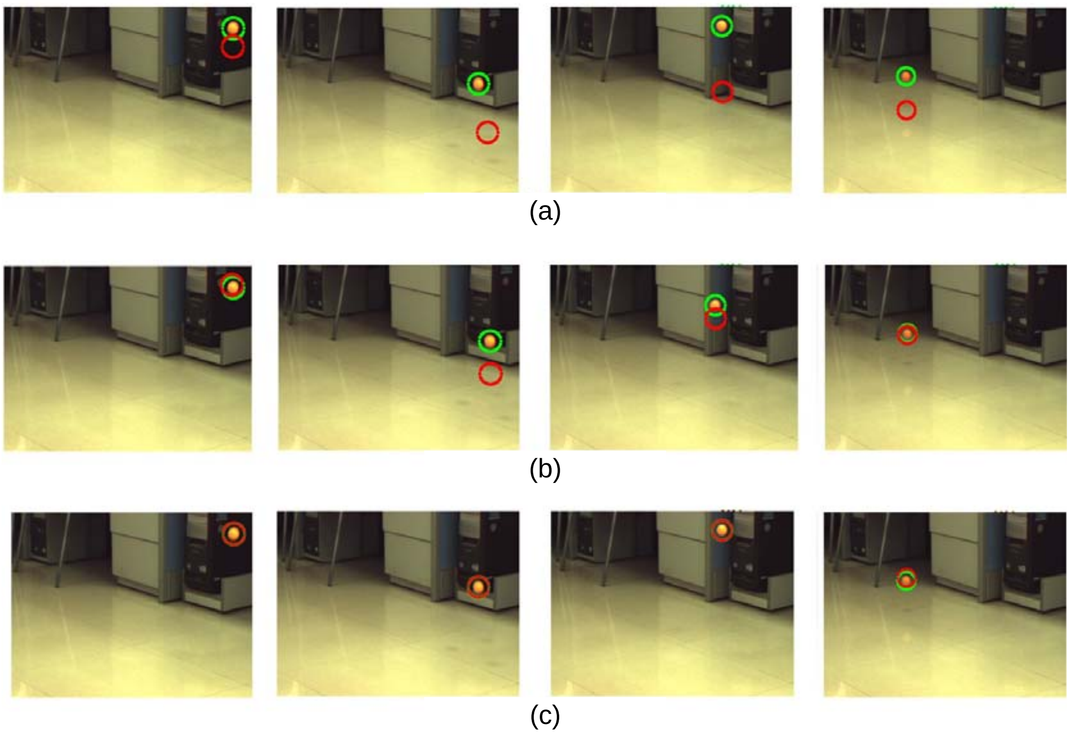


Figure 6. Tracking Performances of the Two Trackers On Dynamic Changes Video (green target position, red tracking result), (b) SIRBF with 150 particles, (c) SIRBF with 1000 particles, (d) our NBRF with 150 particles.

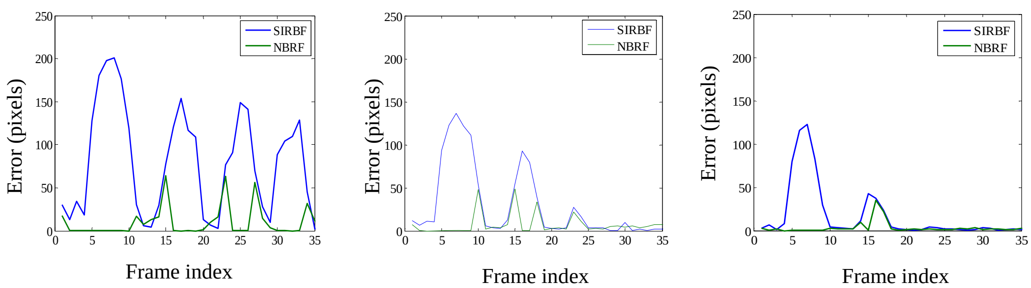


Figure 7. The Tracking Position Error in Pixels of the SIRBF and the NBRF with 100,500 ,1000 particles, respectivel

Scenario 3 sudden dynamic changes with low-frame-rate video: The final 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 NBRF and

SIRBF. The resulr of sample frames are illustrated in Figure 8. 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 our NBRF is better than the SIRBF, and it also illustrate the neural network plays an important role in error compensation to improve the NBRF tracking ability, experiments proved that our proposed NBRF with intensive tracking performance for largely unconstrained abrupt motion even with less number of particles.

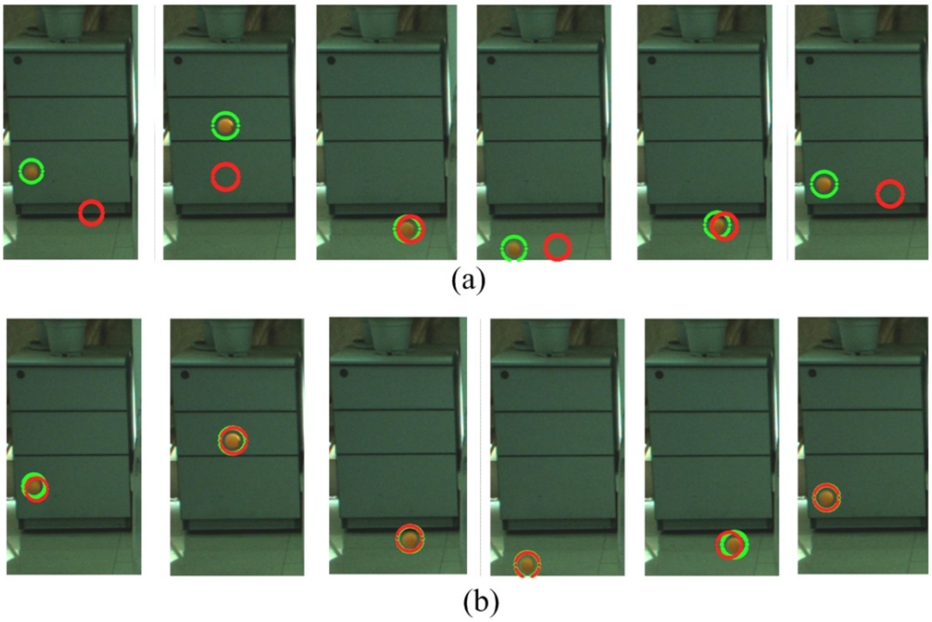


Figure 8. Compare the Tracking Results of SIRBF and NBRF with the smae100 Particles on the Sudden Dynamic changes with Low-frame-rate video (green target position, red tracking result)

For the sake of test our proposed tracker's robustness for different number of particle, many other tracking experiments with the video same as Figure 8, 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 our 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					
	50	100	300	500	1000	>1000
SIRBF	lost	0.41	0.57	0.65	0.81	0.83
NBRF	0.66	0.87	0.96	0.96	0.96	0.96

5. Conclusion

In this paper, a method with neural network auxiliary seqential importance resampling Bayesian filtering (SIRBF) has been presented to improve the performance for robust tracking with lower particles. The SIRBF can enlarge the sampling variance to cover the possible motion uncertainty, however, the high computational burden caused by a large number of particles often makes the SIRBF infeasible for real-time applications. Therefore, a novel method was proposed with RBFNN merge together with resampling-based Bayesian filtering, and the RBFNN has useful to improve the tracking precise even with less particles. Many compare experiments illustrate that our proposed 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.

Acknowledgements

This work is supported by National Science Foundation of China under Grant 61305117.

References

[1] SY Chen. Kalman Filter for Robot Vision: A Survey. *IEEE Transactions on Industrial Electronics*. 2012; 59(11): 4409-4420.
[2] G Gualdi, A Prati, R Cucchiara. Multistage Particle Windows for Fast and Accurate Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2012; 34(8): 1589-1604.
[3] Rafael Muñoz-Salinas, R Medina-Carnicer, FJ Madrid-Cuevas. A Carmona-Poyato. People detection and tracking with multiple stereo cameras using particle filters. *Journal of Visual Communication and Image Representation*. 2009; 20(5): 339-350.
[4] J Anthony Brown, David W Capson. A framework for 3D model-based visual tracking using a GPU-accelerated particle filter. *IEEE Transactions on Visualization and Computer Draphics*. 2012; 18(1): 68-80.
[5] L Bai, Y Wang. Road tracking using particle filters with partition sampling and auxiliary variables. *Computer Vision and Image Understanding*. 2011; 115(10): 1463–1471.
[6] W Kun, W Jie, S Tao. *Improvement of fission bootstrap particle filtering*. 10th International Conference on Electronic Measurement & Instruments (ICEMI). Chengdu. 2011; 3: 94-100.
[7] Z Khan, T Balch, F. Dellaert. *An MCMC-based particle filter for tracking multiple interacting targets*. In Proc. ECCV. Prague. 2004; 3024: 279-290.
[8] K Choo, D Fleet. *People tracking using hybrid Monte Carlo filtering*. In Proc. ICCV. Vancouver, BC. 2001; 2: 321–328.
[9] Y Atchad, JS Rosenthal. On adaptive Markov chain Monte Carlo algorithms. *Bernoulli*. 2005; 11(5): 815-828.
[10] GO Roberts, JS Rosenthal. Coupling and ergodicity of adaptive Markov chain Monte Carlo algorithms. *Journal of applied probability.* 2007; 44(2): 458-475.
[11] Xiuzhuang Zhou, Yao Lu, Jiwen Lu, Jie Zhou. Abrupt Motion Tracking Via Intensively Adaptive Markov-Chain Monte Carlo Sampling. *IEEE Transactions on image processing*. 2012; 21(2): 789-801.
[12] F Liang, C Liu, RJ Carroll. Stochastic approximation in Monte Carlo computation. *Journal of American Statistical Association.* 2007; 102(477): 305-320.
[13] M Isard, A Blake. Condensation-conditional density propagation for visual tracking. *International Journal of Computer Vision*. 1998; 29(1): 5-28.
[14] AC Sankaranarayanan, A Srivastava, R Chellappa. Algorithmic and architectural optimizations for computationally efficient particle filtering. *IEEE Transactions on Image Processing*. 2008; 17(5): 737-748.
[15] Y Wu, J Lin, T Huang. Analyzing and Capturing Articulated Hand Motion in Image Sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2005; 27(12): 1910-1922.
[16] M.Bray, E Koller-Meier, L Van Gool. Smart particle filtering for high-dimensional tracking. *Computer Vision and Image Understanding*. 2007; 106(1): 116-129.