Effect of Resampling Steepness on Particle Filtering Performance in Visual Tracking

Zahidul Islam, Chi-Min Oh, and Chil-Woo Lee
School of Electronics and Computer Engineering, Chonnam National University, South Korea

Abstract: *This paper presents a proficiently developed resampling algorithm for particle filtering. In any filtering algorithm adopting the perception of particles, especially in visual tracking, resampling is an essential process that determines the algorithm's performance and accuracy in the implementation step. It is usually a linear function of the weight of the particles, which determines the number of particles copied. If we use many particles to prevent sample impoverishment, however, the system becomes computationally too expensive. For better real-time performance with high accuracy, we introduce a Steep Sequential Importance Resampling (S-SIR) algorithm that can require fewer highly weighted particles by introducing a nonlinear function into the resampling method. Using our proposed algorithm, we have obtained very remarkable results for visual tracking with only a few particles instead of many. Dynamic parameter setting boosts the steepness of resampling and reduces computational time without degrading performance. Since resampling is not dependent on any particular application, the S-SIR analysis is appropriate for any type of particle filtering algorithm that adopts a resampling procedure. We show that the S-SIR algorithm can improve the performance of a complex visual tracking algorithm using only a few particles compared with a traditional SIR-based particle filter.*

Keywords: *Resampling, particle filter, multi-part colour histogram, steepness parameter, object tracking.*

Received December 27, 2010; accepted March 1, 2011

1. Introduction

Particle filtering is a Sequential Monte Carlo (SMC) method that has demonstrated strong potential for signal- and image-processing applications. It performs three opera-tions sequentially: generating new particles sampling step, computing particle weights importance step and resampling [13]. More specifically, a particle filter is a combination of two elements: Sequential Importance Sampling (SIS) [4, 13] and resampling. This combination of SIS and resampling is called Sequential Importance Resampling (SIR). In the SIS algorithm, after multiple iterations, only very few particles have non-zero importance weights. This phenomenon is often described as weight degeneracy or sample impoverishment. An intuitive solution is to multiply the particles with high normalized importance weights and discard those with low normalized importance weights, which can be done in the resampling step. In practice, however, current resampling algorithms cannot really prevent weight degeneracy; they only reduce the calculation time by discarding particles associated with insignificant weights. In the proposed Steep Sequential Importance Resampling (S-SIR) algorithm, we change the conventional resampling principle of SIR by using a nonlinear function that attenuates particles and uses fewer more effective and higher-weighted particles. The steepness parameter in S-SIR can control the number of the best particles on the basis of weight.

Resampling usually but not necessarily occurs between two importance sampling steps. It can be performed at every step or only if it is regarded as necessary. In our proposed S-SIR algorithm, the resampling schedule has been chosen deterministically instead of dynamically. In a deterministic framework, resampling is done at every k time steps (usually k=1). In a dynamic schedule, a sequence of thresholds (varying time constant) is established and the variance of the importance weight is monitored; resampling is done only when the variance is over the threshold. The strength of the resampling step in the SIS algorithm has been verified by many researchers, as described in [11], but since it also causes additional variation, additional adjustments are needed.

Furthermore, the performance of a tracking system depends greatly on the target object representation and the similarity measurement between the target and the reference object, which is called the measurement model or the observation model. Most of the proposed tracking algorithms are application dependent [2, 5, 17]. Many rely on a single cue, for example, color, which can be chosen according to the application context. Color-based tracking has some advantages, but there can be disadvantages to having an object in a flat color. An efficient color-based target representation can be made using multiple regions of the color histogram by multiple integral images [14], which is a Multi-Part Histogram (MPH) method; it is very helpful for dealing with occlusions. In this paper, our S-SIR-based object tracking method is driven by an MPH-based measurement technique. The most heavily weighted particles are located in the central region of the target by a weighting function, because the other areas of the target are not as important as the center. The Bhattacharyya coefficient [1] is used as a metric to calculate the similarity of the MPH.

The rest of paper is organized as follows. Section 2 describes work related to our current study. Section 3 presents a brief overview of particle filters and resampling algorithms. Section 4 introduces the proposed S-SIR algorithm. The implementation algorithm is discussed in section 5 with proposed human body descriptor used for tracking with an MPH. The results of experiments using two real-time videos with severe occlusions are discussed in section 6; an evaluation and comparison study are also presented. Concluding remarks are given in section 7.

2. Motivation and Related Work

2.1. Motivation

A SIR-based particle filter tracks multiple hypotheses simultanously; each hypothesis is represented by a sample, called a particle, from a weighted set of hypothesis samples (particles). At time t, this set consists of n object states $x_t^1,...,x_t^n$ and their associated weights $w_t^1,...,w_t^n$. The particle set is a discrerte approximation of the posterior distribution of the real object state given the observations up to time t: $p(x_t|y_{0:t})$. At the next step, the particles are resampled according to their weights. This is done to decrease the number of low-weighted particles and increase the number of particles with high weights. Another reason is that the estimated state is the weighted average of all particles; hence, this function directly affects the estimates. For SIR to be successful, a large number of samples particles N is needed for two reasons:

1. To obtain a good approximation of $p(x_t|y_{0:t})$.
2. To be capable of recovering from object loss and to find multiple instances if more than one object is visible. However, the size N is directly related to the computational cost and should be kept as low as possible. This is the key fact motivating us to use the dynamic steepness parameter in resampling. We have seen how the steepness effect of resampling improves the tracking performance even with a very small N particles.

2.2. Related Studies

To enhance the efficiency of particle filtering for small N, many improvements have been suggested. For example, hierarchical methods [3] using a coarse-to-fine approach are used to find the real modes of objects without getting stuck in local optima. Other methods involve sophisticated resampling and/or prediction [12]. However, little research has focused on improving the resampling function for excellent performance with few particles. By using our proposed steepness parameter in resampling, we can dynamically control the particle number as desired.

Several approaches have been used to improve the resampling strategy in visual tracking. Systematic resampling with an adaptive template for visual tracking has been proposed [16]. Systematic resampling had already been established in [13], and the newer method is still a linear-type function. A sampling strategy aimed at reducing computational complexity in the particle filtering framework has also been proposed [15]. This strategy combines particle filtering with a transition prior and an unscented Kalman filter. Our approach differs in that it is a nonlinear type; it is ideally suited to real-time, highly accurate visual tracking. In this article, we incorporate a nonlinear function into this resampling algorithm so it chooses only a few of the best particles with high weight by reducing the search area. All high-weight particles are concentrated appropriately on the tracked object, reducing the possibility of tracking failure and enhancing performance significantly.

3. Particle Filter Overview

In particle filtering, we want to compute the filtered estimates of x_t that is, $p(x_t|y_t)$, based on the set of all available measurements up to time t. In Bayesian estimation, $p(x_t|y_t)$ is computed recursively, that is, in terms of the posterior density at the previous time step, $p(x_{t-1}|y_{t-1})$. A particle filter algorithm uses a set of weighted samples drawn from the posterior distribution to approximate integrals as discrete sums given a set of N random samples $\{x^i_{1:t-1}, w^i_{1:t-1}\}_{i=1,2,...,N}$, where $w^i_{1:t-1}$ are the respective weights and $y_{1:t-1}$ are the available measurements up to time t. According to the SIS strategy, the posterior distribution can be computed as:

$$p(x_t \mid y_{1:t}) \approx \sum_{i=1}^{N} w_t^i \delta(x_t - x_t^i) \tag{1}$$

where $\delta(.)$ is the Dirac delta function. It is usually impossible to sample from the posterior distribution directly. This matter can be resolved by drawing samples from a proposal distribution $q(x_{1:t-1} \mid y_{1:t-1})$. Choosing the proper proposal distribution is an important step when using an importance sampling algorithm. The most popular choice of proposal distribution is the prior distribution because of its ease of calculation. The proposal distribution can be expressed as:

$$p(x_t \mid x_{t-1}) = q(x_t \mid x_{t-1}, y_t)_{i=1,2,...,N} \tag{2}$$

If the prior distribution is selected as the proposal distribution, the importance weight calculation can be expressed simply as:

$$w_t^i = w_{t-1}^i \frac{p(y_t \mid x_t^i) p(x_t \mid x_{t-1})}{q(x_t \mid x_{1:t-1}^i, y_{1:t})} \quad (3)$$

The mean state of an object is estimated at each time step by:

$$\hat{E}[x_t] = \sum_{i=1}^{N} w_t^i x_t^i \quad (4)$$

4. Nonlinear Resampling Algorithm

In resampling, particles with large weights are replicated, and those with negligible weights are removed. Resampling maps the weighted random measure $\{x_{0:t}^i, \tilde{w}_k(x_{0:t}^i)\}$ onto the equally weighted random measure $\{x_{0:t}^i, N\}$ by sampling uniformly with replacement from sample space on the basis of the probabilities. Many improved particle filters focus on resampling, for instance, that in [10], in which the authors proposed using a partially deterministic reallocation scheme instead of resampling to overcome the extra variation arising in resampling.

We modify the SIR-based particle filter by changing the resampling function to a nonlinear function. In real-time visual tracking, the SIR filter works well, but effective sorting of higher-weight particles in every iteration is computationally expensive, and tracking failure becomes more likely. Our ultimate goal can be divided into two parts. First, we want to use fewer of the best-weighted particles; second, by reducing the number of particles, we want to get the best tracking output. Our proposed method reduces calculation time and uses the lowest number of the highest-weight particles by using a steepness function. The steepness parameter can also control the number of the best particles used, which is mainly application dependent, as desired. The traditional resampling algorithm is a linear mapping function that copies or replaces particles with high weight. It can be expressed as:

$$\Gamma = w_t^i . n \quad (5)$$

Where Γ is the new sorted particles set, w is the relevant weight, and n is the particle number. We can copy the more effective particles by discarding those associated with insignificant weights using:

$$\Xi = a * (exp(b * (w_t^i))) + c \quad (6)$$

where Ξ is new assigned weight with non-linear mapping, b is an attenuating factor (steep parameter), and a and c are arbitrary constants (a, $b \neq 0$). The number of particles copied for resampling can be controlled by the steepness parameter b, as shown in Figure 1. Besides, how we can assign suitable value of b is discussed in the experiment section. Figure 1 shows that this nonlinear mapping helps attenuate particles by discarding low-weight particles, which is better than the linear mapping used in conventional resampling. To normalize equation 6, we can write it as:

$$W_t^i = \frac{\Xi}{\sum_{i=1}^{N} w_t^i} \quad (7)$$

Finally, equation 5 can be rewritten with the help of equation 7 as:

$$\Gamma = round\ (W_t^i . n) \quad (8)$$

However, this straightforward algorithm creates the problem of weight degeneration. Resampling algorithms have been applied to overcome this problem.

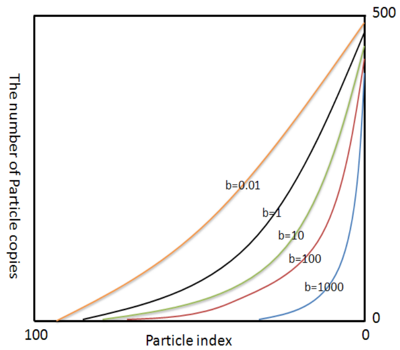


Figure 1. Effect of steepness parameter in resampling based on weight.

To clarify our proposed resampling, we can compare it to the traditional resampling strategy shown in Figure 2, which is discussed in many articles, e.g., in [6, 9, 10, 13, 15, 16]. As we see from Figure 1, the steepness parameter controls the resampled particles nonlinearly, as illustrated in Figure 3. We can boost the steepness by increasing b, and the increased steepness can accumulate the best weighted particles in a more concentrated way. However, increasing b too much may create another problem by losing the basic multi-modality of the particle filter. Thus, we must be careful to choose the best steepness parameter b, which may be application dependent.

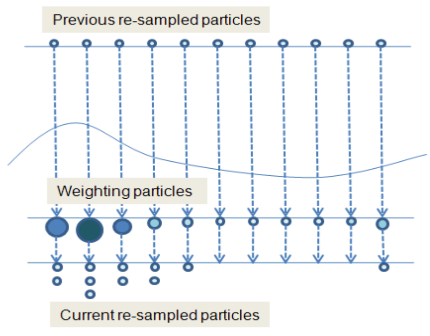


Figure 2. Basic resampling strategy.

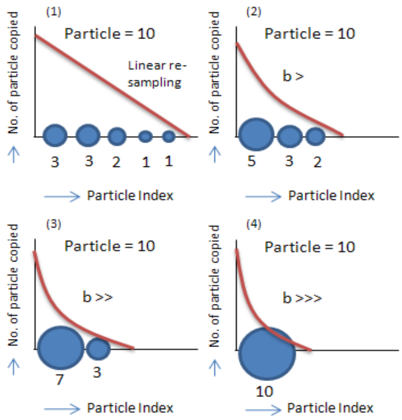


Figure 3. S-SIR based resampling strategy nonlinear.

This example uses a 10-particle simulation. As we increase *b*, the best weighted particle is copied many times by discarding other, lower-weighted particles.

5. Particle Filtering Based Implementation

Particle filtering estimates the proposal distribution using samples from previous posterior distributions. This estimation requires approximation, which is weighted by the observation model. Robust tracking demands a robust observation model.

5.1. Measurement

5.1.1. Object Feature Descriptor

In this paper, the tracked human body is considered to consist of various rectangle regions. We introduce the MPH, which uses an integral-image-based representation [14] that characterizes the human body using detailed spatial information. The details of MPH can be found from our previous paper in [8, 9].

5.1.2. Colour Measurement Model

To achieve robustness against non-rigidity, rotation, and partial occlusion, we focus on the color distributions as target models. The details of this proposed color model also can be found from our previous paper in [8, 9]. We use a multiple-region color histogram as one of the observation measurements to weight the sample set. The observation accuracy depends on the object's features. We adopt a Gaussian density for the likelihood function of the measured color histogram.

5.2. The Motion Model

We model the state location in each frame of a video. The state space is represented in the spatial domain as $X=(x, y)$. The state space for the first frame is initialized manually by selecting the object of interest in a video scene using a rectangle. A second-order autoregressive dynamic is chosen from the parameters used to represent our state space, i.e., (x, y). The dynamic is given as $X_{t+1}=Ax_t+Bx_{t-1}$. Matrices A and B could be determined from a set of sequences in which the correct tracks have been obtained.

5.3. The Observation and Likelihood Model

The observation model we have found using the measurement model in subsection 5.1, which is used to measure the observation likelihood of the samples. This is important in object tracking. The filter corrects the predicted estimation by using the observed data. The overall likelihood calculation based on the MPH is given by:

$$\begin{aligned} L(y_t \mid x_t) &\propto D_t(p, q_t) \\ &= L_{MPH}(y_{MPH,t} \mid x_t) \end{aligned} \tag{9}$$

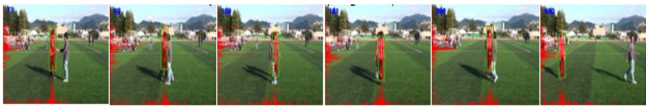
where $D_t = dist[p, q_t]$ is the distance between the reference histogram p of the objects to be tracked and the histogram q_t computed from image in the region defined by the state vector x_t and y_t denotes the measurement vector, which is composed of the measurement vectors $y_{MPH,t}$ from the MPH-based colour cue.

6. Experiments and Results

We verify the performance of our algorithm experimentally using two different video sequences from our own database and another well-known database, CAVIAR [7]; we aim to track a pre-selected moving person. In the first sequence, circleocc (500 frames), two people are walking toward each other from opposite sides of the frame. They meet, shake hands, and circle each other; our subject is completely occluded more than three times. The second sequence, OneShopOneWait2cor (211 frames), was obtained from the CAVIAR [7] database. Another person causes a lengthy occlusion of the target.

Figure 4 compares the tracking performance of our proposed resample-based algorithm with that of the traditional algorithm using 100 and 10 particles for circleocc. For this comparison, we use frames 71, 99, 109, 181, 296, and 480 in all the tests. Our proposed system works well even with only 10 particles, as shown in Figure 4-c. The possibility of tracking failure when using only 10 particles was drastically reduced. The overall performance can be verified by the horizontal and vertical red bars shown in each frame, which represent the probability densities of the estimated state. In Figure 4-b, we can see that the probability densities become scattered to find the best weighted particle for the next state estimation. This works well sometimes but only when using a large number of particles. However the possibility of tracking failure remains owing to the many real-time challenges in visual tracking. In this video, we use steepness parameter b=1000.

a) Proposed new resampling: 100 particles.

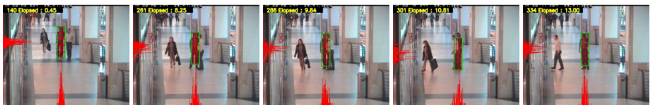


b) Traditional resampling: 100 particles.

c) Proposed new resampling: 10 particles.

d) Traditional resampling: 10 particles.

Figure 4. Tracking in circleocc with traditional and proposed resampling using 100 and 10 particles. Overall performance can be verified by the horizontal and vertical red bars shown in each frame, which represent the probability densities of the estimated state.



a) Proposed new re-sampling: 100 particles.

b) Traditional SIR based re-sampling: 100 particles.

c) Proposed new re-sampling: 10 particles.

d) Traditional SIR based re-sampling: 10 particles.

Figure 5. Tracking in OneShopOneWait2cor with traditional and proposed resampling using 100 and 10 particles. Overall performance can be verified by the horizontal and vertical red bars shown in each frame, which represent the probability densities of the estimated state.

Figure 5 compares the performance of our proposed resampling method and a traditional SIR-based resampling algorithm with 100 and 10 particles for OneShopOneWait2cor. In this experiment, we use frames 140, 261, 288, 301, and 334 in all tests. As with the performance illustrated in Figure 4, this OneShopOneWait2cor video sequence also verifies the effectiveness of our proposed system. Using 10 particles, the traditional SIR-based resampling fails to track the target, as shown in Figure 5-d, whereas our proposed system works very well, as shown in Figure5-c. In both cases, we use steepness parameter b=500. A more quantitative result is discussed in the following sections.

6.1. The Error Metric

In the previous section, our proposed system was evaluated qualitatively. The Root Mean Squared Error (RMSE) method in the state space has also been used to evaluate the performance of our algorithm. The RMSE can be formulated by:

$$RMSE(t) = \sqrt{0.5((g_t - \hat{g}_t)^2 + (h_t - \hat{h}_t)^2)} \tag{10}$$

where $(\hat{g}_t, \hat{h}_t)$ stands for the upper-left corner coordinates of the tracking box determined by the central position corresponding to the state estimated by the particle filter in the frame. The ground truth states (g_t, h_t) correspond to the true positions of the object and have been generated by manually creating a tracking box surrounding the object in the test videos.

6.2. The Performance Evaluation

We evaluate our proposed system with 100 and 10 particles using different steepness parameters to observe the tracking output of the two example videos. The RMSE graph for the circleocc video stream with different steepness parameters is shown in Figures 6 and 7. When b=1 and 100, the tracking performance is not as good as desired, and it is better when b=500 to 5000, remaining almost constant in that range. A more numerical analysis is given in Table 1.

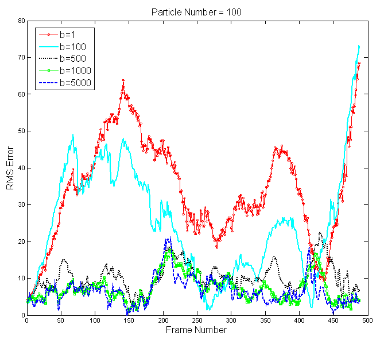


Figure 6. Performance of proposed resampling for circleocc video stream with different steepness parameters (b=1 to 5000) using 100 particles.

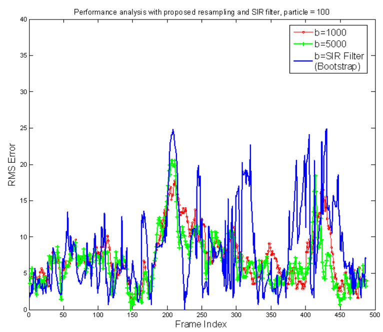


Figure 7. Performance of proposed S-SIR method and conventional SIR filter.

Table 1. Performance of proposed resampling and SIR filter for circleocc video stream with different steepness parameters using 100 particles.

Steep Par.b	1	100	500	1000	2000	5000	Trad. SIR
Max RMSE	68.43	73.25	22.47	17.73	18.5	20.5	24.95
Min RMSE	4.43	1.41	2.82	1	0.7	0	0
Avg. RMSE	34.25	25.3	10.3	7.2	6.8	6.5	8.15
P	100	99	95	87	75	57	N/A

We also compared our results with those of a conventional SIR filter. Our proposed S-SIR performed much better than the SIR. For example, when b=1000, the maximum RMSE and average error are 17.73 and 7.2, respectively. In contrast, for the conventional SIR, the maximum RMSE and the average error are 24.95 and 8.15, respectively. Also, the last row of this Table shows the number of best particles used (p) at different steepness parameters b. As we increase b, the number of particles used decreases. This reduces the calculation time, making our system faster. However, if we increase the steepness parameter too much, the particle filter may lose its multi-modality, which creates another problem.

We chose the optimum steepness on the basis of the tracking environment and the desired result. The graph in Figure 7 compares the performance of the proposed S-SIR method and a conventional SIR filter. Our proposed algorithm works well with only 10 particles, whereas the SIR-based particle filter totally fails to rack the object. The graph in Figure 8 shows the tracking performance with different steepness factors with only 10 particles. That in Figure 9 compares the performance of the proposed S-SIR and a SIR-based particle filter with 10 and 100 particles. Table 2 summarizes the RMSE at different steepness parameters with only 10 particles. The last column of this Table also shows the corresponding tracking performance of a conventional SIR-based particle filter.

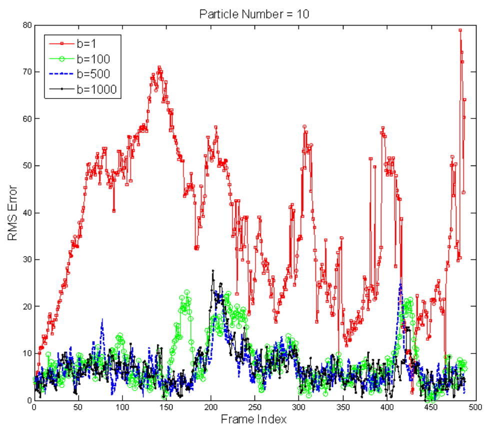


Figure 8. Performance of proposed resampling for circleocc video stream with different steepness parameters b(b= 1 to 1000) using 10 particles.

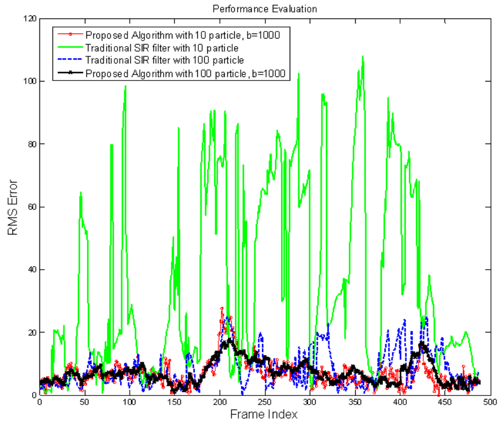


Figure 9. Performance of proposed S-SIR and SIR with 100 and 10 particles.

Table 2. Performance of proposed resampling and a SIR filter for circleocc video stream with different steepness parameters using 10 particles.

Steep Par. b	1	100	500	1000	2000	5000	Trad. SIR
Max RMSE	78.93	23.1	24.7	27.6	29.2	27.7	107.9
Min RMSE	1.6	0	0	0	0.7	0	0.7
Avg. RMSE	36.15	8.2	6.87	6.82	7.0	6.78	36.2
P	10	9	6	4	3	2	N/A

Table 3. Performance of proposed resampling and SIR filter for OneShopOneWait2cor video stream with different steepness parameters using 100 particles.

Steep Par. b	1	100	500	1000	2000	5000	Trad. SIR
Max RMSE	25.01	24.7	13.15	20.02	23.02	32.4	26.17
Min RMSE	1.41	4.47	0	0	0	0	0
Avg. RMSE	13.01	18.2	2.87	3.12	3.95	3.5	4.3
P	100	99	84	71	56	37	N/A

The RMSE graph for the OneShopOneWait2cor video stream at different steepness parameters is shown in Figures 10 and 11. Figure 10 shows the parformance using 100 particles with different steepness parameters, and Figure 11 shows the performance using only 10 particles. In both cases, the best-tuned steepness parameter is b=500. Tables 3 and 4 summarize the RMSE at different steepness parameters with 100 and 10 particles, respectively. The last column of these Tables also shows the corresponding tracking performance of a conventional SIR-based particle filter. For example, when b=500, the maximum RMSE and average error are 13.15 and 2.87, respectively. In contrast, with a conventional SIR, the maximum RMSE and average error are 26.17 and 4.3, respectively. As noted above, Table 4 illustrates that, with 10 particles, the SIR-based filter totally fails to track the object. For the SIR-based filter with 10 particles, the maximum and average error are 240.5 and 96.9, respectively whereas with our

proposed system, they are 11.4 and 3.92, respectively (steepness parameter b=500). Note that we must carefully tune the steepness parameter, as we do not want to lose the multi-modality of the particle filter. This may happen if we increase the steepness parameter too much (more than 10,000).

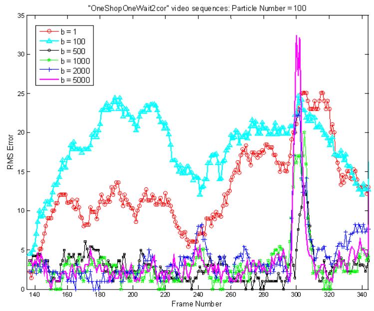


Figure 10. Performance of proposed resampling for OneShopOneWait2cor video stream with different steepness parameters b using 100 particles (b=1 to 5000).

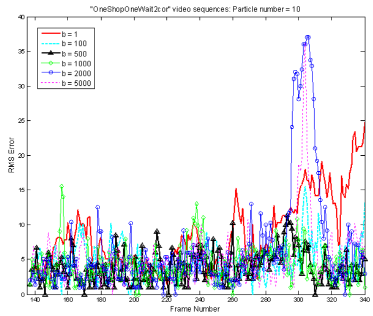


Figure 11. Performance of proposed resampling for OneShopOneWait2cor video stream with different steepness parameters b using 10 particles (b=1 to 5000).

Table 4. Performance of proposed resampling and SIR filter for OneShopOneWait2cor video stream with different steepness parameters using 10 particles.

Steep Par. b	1	100	500	1000	2000	5000	Trad. SIR
Max RMSE	24.75	15.52	11.4	15.52	37	36.12	240.5
Min RMSE	1.41	1.41	0	0	0	0	2.23
Avg. RMSE	8.63	8.63	3.92	4.37	6.58	5.11	96.9
P	10	7	4	3	2	1	N/A

7. Conclusions

A new particle filter with S-SIR-based resampling has been proposed in this paper. The proposed resampling design addresses problems with efficient control of the best weighted particles by attenuation. The steepness factor can control the number of high-weight particles as desired. It also reduces the calculation time during tracking, and even a few particles can provide satisfactory tracking. This S-SIR algorithm improves the object-tracking performance compared to a conventional SIR-based filter. The proposed nonlinear type of resampling can determine the most important particles and attenuate the other particles very efficiently. Also, from the experimental results, we can conclude that this proposed algorithm minimizes the degradation of real-time performance and remarkably reduces the computational complexity. With our optimally tuned steepness parameter, we can obtain our desired tracking performance with few particles, whereas a conventional system cannot track at all with a small number of particles.

Acknowledgements

This research was supported by The Ministry of Knowledge Economy (MKE), Korea, partially under the Information Technology Research Center (ITRC) support program and partially under the human resources development program for Convergence Robot Specialists support program supervised by the National IT Industry Promotion Agency (NIPA) (NIPA-2010-C1090-1011-0008) and (NIPA-2010-C7000-1001-0007).

References

[1] Aherne F., Thacker N., and Rockett P., "The Bhattacharyya Metric as an Absolute Similarity Measure for Frequency Coded Data," *Kybernetika*, vol. 34, no. 4, pp. 363-368, 1997.

[2] Artuar L., Lyudmila M., and David B., "Structural Similarity-Based Object Tracking in Multimodality Surveillance Videos," *Machine Vision and Applications*, vol. 20, no. 2, pp. 71-83, 2009.

[3] Deutscher J., Blake A., and Ried I., "Articulated Body Motion Capture by Annealed Particle Filtering," *in Proceedings of Computer Vision and Pattern Recognition*, USA, vol. 2, pp. 126-133, 2000.

[4] Douc R. and Cappe O., "Comparison of Resampling Schemes for Particle Filtering," *in Proceedings of the 4th International Symposium on in Image and Signal Processing and Analysis*, pp. 64-69, 2005.

[5] Ganesan K. and Jalla S., "Video Object Extraction Based on a Comparative Study of Efficient Edge Detection Techniques," *The International Arab Journal of Information Technology*, vol. 6, no. 2, pp. 107-115, 2009.

[6] Green P., "Reversible Jump Markov Chain Monte Carlo Computation and Bayesian Model Determination," *Biometrika*, vol. 82, no. 4, pp. 711-732, 1995.

[7] Homepages, available at: http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1, last visited 2011.

[8] Islam M., Oh C., and Lee C., "An Efficient Multiple Cues Synthesis for Human Tracking Using A Particle Filtering Framework," *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 6, pp. 3379-

3393, 2011.

[9] Islam M., Oh C., and Lee C., "Modifed Re-Sampling Based Particle Filter for Visual Tracking with MPH," *in Proceedings of 10th IEEE International Conference on Computer and Information Technology*, Bradford, pp.1582-1587, 2010.

[10] Liu J., Chen R., and Logvinenko T., *A Theoretical Framework for Sequential Importance Sampling with Resampling*, Sequential Monte Carlo Methods in Practice, Eds., Springer Verlag, Berlin, 2001.

[11] Liu J. and Chen R., "Blind Deconvolution Via Sequential Imputation," *Journal of American Statistical Association*, vol. 90, no. 430, pp. 567-576, 1995.

[12] Pitt M. and Shephard N., "Filtering Via Simulation: Auxiliary Particle Filters," *Journal of the American Statistical Association*, vol. 94, no. 446, pp. 590-599, 1999.

[13] Sanjeev A., Simon M., Neil G., and Tim C., "A Tutorial on Particle Filters for Online nonlinear/non-gaussian Bayesian Tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174-188, 2002.

[14] Viola P. and Jones M., "Rapid Object Detection Using A Boosted Cascade of Simple Features," *in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. I-511-I-518, 2001.

[15] Wang F. and Lin Y., "Improving Particle Filter with A New Sampling Strategy," *in Proceedings of 4th International Conference on Computer Science and Education*, Nanning, pp. 408-412, 2009.

[16] Wu G. and Tang Z., "A New Resampling Strategy about Particle Filter Algorithm Applied in Monte Carlo Framework," *in Proceedings of Second International Conference on Intelligent Computation Technology and Automation*, Hunan, pp. 507-510, 2009.

[17] Yunqiang C. and Yong R., "Real Time Object Tracking in Video Sequences, Signals and Communications Technologies," *Interactive Video*, Springer Berlin Heidelberg, Part 2, pp. 67-88, 2006.

Zahidul Islam received his BSc and MSc degrees from the Department of Applied Physics and Electronic Engineering, University of Rajshahi, Bangladesh, in 2000 and 2002 respectively. In 2003, he joined as a lecturer in the Department of Information Communication Engineering, Islamic University, Bangladesh. He is currently, working on the development of visual object tracking system with various aspects as a PhD candidate of computer engineering in intelligent image media and interface lab, Chonnam National University, South Korea. His other current research interests include computer vision, 3D object, human and motion tracking and tracking articulated body, genetic algorithm etc.

Chi-Min Oh received his BSc and MSc degree in computer engineering from Chonnam National University, Gwang-ju, Korea in 2007 and 2009 respectively. Since February 2009, he has been pursuing the PhD degree in School of Electronics and Engineering, Chonnam National University. His research interests include gesture recognition and articulated body tracking.

Chil-Woo Lee received his BSc and MSc degrees in electronic engineering from Chung-Ang University in 1986 and 1988 respectively in Seoul, Korea. And he received PhD also in electronic engineering in 1992 from University of Tokyo, Japan. Since 1996, he has been a professor, Deptment of Computer Engineering, Chonnam National University in Korea. He has worked as senior researcher at laboratories of image information science and technology for four years, form 1992 to 1996, and at that time he had an extra post of visiting researcher at Osaka University in Osaka, Japan. From January, 2001, he has visited North Carolina A and T University as a visiting researcher and jointly worked on several digital signal processing projects. He is now a director of two research institutes; mobile device research centre and culture technology research institute, and those are financially supported by the government of Rep. Korea. Up to now, his research work has been associated with image recognition and image synthesis. His research interests include computer vision, computer graphics, and visual human interface system. And he is also very interested in realization of real-time sensor system that can be aware of context of circumference.