Tracking of Abrupt Motion Using Wang-Landau Monte Carlo Estimation

Junseok Kwon and Kyoung Mu Lee

Department of EECS, ASRI, Seoul National University,151-742, Seoul, Korea
paradis0@snu.ac.kr, kyoungmu@snu.ac.kr

Abstract. We propose a novel tracking algorithm based on the Wang-Landau Monte Carlo sampling method which efficiently deals with the abrupt motions. Abrupt motions could cause conventional tracking methods to fail since they violate the motion smoothness constraint. To address this problem, we introduce the Wang-Landau algorithm that has been recently proposed in statistical physics, and integrate this algorithm into the Markov Chain Monte Carlo based tracking method. Our tracking method alleviates the motion smoothness constraint utilizing both the likelihood term and the density of states term, which is estimated by the Wang-Landau algorithm. The likelihood term helps to improve the accuracy in tracking smooth motions, while the density of states term captures abrupt motions robustly. Experimental results reveal that our approach efficiently samples the object's states even in a whole state space without loss of time. Therefore, it tracks the object of which motion is drastically changing, accurately and robustly.

1 Introduction

Object tracking is a well known problem to computer vision community [1]. Visual tracking has been utilized in surveillance systems and other intelligent vision systems. Recently, many of the visual tracking systems trends have been towards addressing complex outdoor videos rather than lab environmental ones. These complex outdoor videos which can be easily found in web sites, usually contain drastically abrupt motions.

Traditional tracking methods can be divided into two categories: the sampling based method (stochastic approach) and the detection based method (deterministic approach). In the stochastic approach, the particle filter (PF) has shown efficiency in handling non-gaussianity and multi-modality [2,3]. In multi-object tracking, Markov Chain Monte Carlo (MCMC) reduces computational costs to deal with high-dimensional state space [4,5]. Data-Driven MCMC provides quick convergence results with efficient proposals [6]. The stochastic approach has an advantage of reflecting the motion's uncertainty. In the deterministic approach, the Adaboost detector has been widely used in detecting a target object [7] and various data association techniques have been applied to connect the detected target and make a trajectory [8]. The deterministic approach usually provides reliable results by utilizing the bottom-up information. In general, both these

D. Forsyth, P. Torr, and A. Zisserman (Eds.): ECCV 2008, Part I, LNCS 5302, pp. 387–400, 2008.

(a) Frame #246 (b) Frame #247 (c) Frame #248 (d) Frame #249

Fig. 1. Example of an abrupt motion. The camera shot change causes the boxer to have an abrupt motion at consecutive frames, (b)-(c).

two tracking approaches basically assume that the appearance and motion of an object are smoothly changed over time.

However, in many complex outdoor scenarios, these motion and appearance smoothness constraints are frequently violated. Recently, online feature selection techniques have started to tackle this problem [9,10,11]. New features are selected online to adapt abrupt changes in appearance. Yet, most of tracking methods rarely consider abrupt motions which cause traditional algorithms to fail. In this study, we address the problem of tracking objects whose motion is mostly smooth, but which changes rapidly over one or more small temporal intervals. This motion typically occurs in two challenging situations: (1) video consists of edited clips acquired from several cameras (shot change), (2) object or camera rapidly moves. Figure 1 illustrates an example of the first situation.

The philosophy of our method is that two kinds of the motion, which is smooth or abrupt, can be efficiently tracked at the same time by trading off two factors which are the likelihood term and the density of states term. If the likelihood term is highly weighted, our method is similar with conventional tracking methods which track the smooth motion. On the other hand, if the density of states term is highly weighted, the method has the similar properties of detection methods which could capture the abrupt motion. So, as trading off these two terms, our method combines merits of tracking methods with ones of detection methods.

The first contribution of this paper is that, to the best of our knowledge, we firstly introduce the Wang-Landau Monte Carlo (WLMC) sampling method to the tracking problem. The WLMC sampling method was recently proposed in the statistical physics literature, which accurately estimates the density of states [12]. The second contribution is to propose the WLMC based tracking method and provide the unified framework to track both smooth and abrupt motion without loss of time. In the unified framework, the method utilizes the efficient sampling schedule. The schedule encourages to sample less-visited regions of the state space, while spending more time refining local maxima. And the method provides a statistical way to reach the global maximum. The third contribution is that, in order to design more efficient sampling scheme, we modify the WLMC sampling method into an annealed version and present the annealed WLMC based tracking method. The method searches for interesting subregions of the state space by employing the density of states and reduces the state space to these subregions where the target exists.

2 Related Works

The quasi-random sampling addresses the problem of tracking pedestrians from a moving car [13]. To cope with the abrupt changes in motion and shape, the method combines particle filter with quasi-random sampling. This algorithm has two drawbacks. First, the method chooses highly weighted particles and densely samples new states around the states of those particles. However, if there are a few deeper local maxima, most of samples get trapped in those local maxima. Second, the method uses uniform sampling over the entire state space to capture the abrupt changes. However, if entire state space is very large, uniform sampling scheme can be wasteful.

The cascade particle filter addresses tracking in low frame rate videos [14]. In this approach, the detection algorithm is well combined with particle filter to deal with abrupt motions. It demonstrates efficiency in a face tracking case. However, this approach requires complex observers and an offline training process. On the other hand, we consider the human as a target object, which makes it more challenging than the face, and treat much larger areas in an image for tracking.

3 Wang-Landau Monte Carlo Algorithm

The density of states is the number of states which belongs to a given energy subregion. Let us consider the 2D Ising model and assume that the energy function of the model is defined by only pairwise term [12]. Then, at the lowest energy, the density of states is 2 because the states which yield the lowest energy, are following 2 cases; all nodes have same values, -1 or +1. As it is intractable to accurately calculate the density of states in all energy subregions, the WLMC method [12] approximately estimate the density of states through a Monte Carlo simulation. Let us assume that the energy space $\mathbf{E}$ is divided into d disjoint subregions such that

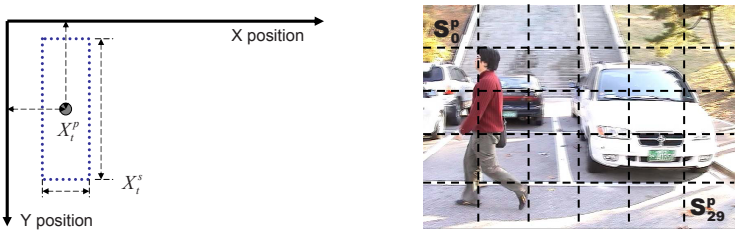
$$E_i \cap E_j = \phi \quad for \quad i \neq j,\ i,j \in \{1,\ldots,d\} \quad and \sum_{all\ i}^{d} E_i = \mathbf{E}. \tag{1}$$

Each energy subregion is visited via random walk in the energy space. If E_i is visited, we increase the histogram $h(E_i)$ by one and modify the density of states $g(E_i)$ by a modification factor $f > 1$. One proper modification method is

$$g(E_i) \leftarrow g(E_i) * f, \tag{2}$$

where $g(E_i)$ is initially set to 1 for all i and gradually updated by (2). While the simulation progresses, the random walk produces a flat histogram over the energy space. Note that flat has a different meaning; As described in [12], if the lowest bin of the histogram is larger than 80% of the histogram average, we consider the histogram is flat. Since the flat histogram means that all energy subregions are visited at least to some degree, the algorithm proceeds to the next random walk in a coarse-to-fine manner.

$$f \leftarrow \sqrt{f}. \tag{3}$$



(a) Example of (X_t^p, X_t^s) (b) Example of S_i^p

Fig. 2. Example of a state and subregion. (a) X_t^p represents the center of an object and X_t^s indicates the size of the boundary box. (b) $\mathbf{S}^p$ is divided into 30 equal-size subregions.

The modification factor is reduced to a finer version by (3), and the histogram is reset. Simulation continues until the histogram becomes flat again and restarts with a finer modification factor. The algorithm is terminated when the factor becomes highly close to 1 or the number of iterations reaches a predefined value.

In the WLMC simulation, a new state is proposed at each time. This new state is accepted or rejected according to the transition probability. The transition probability of the current state from E_i to E_j is defined by

$$p(E_i \rightarrow E_j) = \min\left[1, \frac{1/g(E_j)}{1/g(E_i)}\right]. \tag{4}$$

Note that the transition probability is calculated with the inverse of the density of states. This means that the transition is guided to less visited energy subregions.

4 WLMC Based Tracking Method

4.1 Preliminary

The state $\mathbf{X}_t$ at time t consists of the position and scale of an object; $\mathbf{X}_t = (X_t^p, X_t^s)$. And the state space $\mathbf{S}$ is defined by a set of all possible states. This state space $\mathbf{S}$ can be decomposed into the state space of position and scale; $\mathbf{S} = \mathbf{S}^p \times \mathbf{S}^s$. As, in many cases, an abrupt motion occurs by the change of the position, we assume that the scale of an object is smooth over time and only consider abrupt changes in $\mathbf{S}^p$. $\mathbf{S}^p$ is then divided into d disjoint subregions; $S_i^p, i = \{1, \ldots, d\}$. A simple dividing strategy is to partition $\mathbf{S}^p$ into equal-size grids as shown in Figure 2(b). Note that, to adapt the WLMC method to an image-based tracking problem, we replace E_i in Section 3 with S_i^p and calculate the density of states at each S_i^p. Our method can be easily extended to deal with abrupt changes of the scale by calculating the density of states at $\mathbf{S}^s$.

4.2 Bayesian Object Tracking Approach

The object tracking problem is usually formulated as the Bayesian filtering. Given the state of an object at time t, $\mathbf{X}_t$ and the observation up to time t,

$\mathbf{Y}_{1:t}$, the Bayesian filter updates the posteriori probability $p(\mathbf{X}_t|\mathbf{Y}_{1:t})$ with the following rule:

$$p(\mathbf{X}_t|\mathbf{Y}_{1:t}) = cp(\mathbf{Y}_t|\mathbf{X}_t) \int p(\mathbf{X}_t|\mathbf{X}_{t-1})p(\mathbf{X}_{t-1}|\mathbf{Y}_{1:t-1})d\mathbf{X}_{t-1}, \tag{5}$$

where $p(\mathbf{Y}_t|\mathbf{X}_t)$ is the observation model that measures the similarity between the observation at the estimated state and the given model; $p(\mathbf{X}_t|\mathbf{X}_{t-1})$ is the transition model which predicts the next state $\mathbf{X}_t$ based on the previous state $\mathbf{X}_{t-1}$, and; c is the normalization constant. The observation model generally utilizes color, edges or texture as a feature [1]. With the posteriori probability $p(\mathbf{X}_t|\mathbf{Y}_{1:t})$ computed by the observation model and the transition model, we obtain the Maximum a Posteriori (MAP) estimate over the N number of samples at each time t.

$$\mathbf{X}_t^{MAP} = \underset{\mathbf{X}_t^n}{\arg\max}\, p(\mathbf{X}_t^n|\mathbf{Y}_{1:t}) \quad for\ n = 1, \ldots, N, \tag{6}$$

where $\mathbf{X}_t^{MAP}$ denotes the best configuration which can explain the current state with the given observation. However note that the integration in (5) is unfeasible in high dimensional state space. To address this problem, we use the Metropolis Hastings (MH) algorithm that is one of the popular MCMC method. The MH algorithm consists of two main steps; proposal step and acceptance step.

In this work, the traditional transition model $p(\mathbf{X}_t|\mathbf{X}_{t-1})$ is reinforced by the approximated prior term $p^*(\mathbf{X}_t)$ to track the abrupt motion. Then our transition model is defined by

$$p(\mathbf{X}_t|\mathbf{X}_{t-1}) \approx p(\mathbf{X}_t|\mathbf{X}_{t-1})\frac{p^*(\mathbf{X}_t)}{p(\mathbf{X}_t)} = cp(\mathbf{X}_t|\mathbf{X}_{t-1})p^*(\mathbf{X}_t), \tag{7}$$

where the inverse of the prior term $p(\mathbf{X}_t)$ is replaced with constant c. We sequentially estimate the approximated prior term $p^*(\mathbf{X}_t)$ using the density of states that is calculated by the Wang-Landau recursion step.

4.3 Proposal Step

The proposal density designs the transition from a given state to a new state based on some prior knowledge about the motion. Our prior knowledge on a motion is that objects can go anywhere in a scene even at one proposal step. With this assumption, we design a new proposal density that covers the whole state space and proposes highly diverse states.

$$Q(\mathbf{X}_t'; \mathbf{X}_t) = \begin{cases} Q_{AR}(X_t^{s'}; X_t^s) & \text{for the scale} \\ Q_U(X_t^{p'}) & \text{for the position} \end{cases}. \tag{8}$$

Q_{AR} proposes a new scale state $X_t^{s'}$ based on X_t^s with a second-order autoregressive process. This process well describes the characteristic of the smooth

change in scale [2], and fits our smoothness assumption of the scale. To propose a new position state, Q_U utilizes two steps. The first step randomly chooses one subregion S_i^p to obtain diverse states which cover abrupt changes in position. And the second step uniformly proposes a new state $X_t^{p'}$ over the chosen S_i^p to simulate the density of states at S_i^p.

For the success of our proposal step, its efficiency has to be considered. The proposal density (8) can be wasteful if it proposes numerous inefficient states where the probabilities that the target exists are very low. Hence, our algorithm addresses this inefficiency utilizing the density of states in the acceptance step.

4.4 Acceptance Step

The acceptance step determines whether the proposed state is accepted or not and can be simply calculated by the likelihood ratio between the current and proposed states as follows.

$$a = min\left[1, \frac{p(\mathbf{Y}_t|\mathbf{X}'_t)Q(\mathbf{X}_t;\mathbf{X}'_t)}{p(\mathbf{Y}_t|\mathbf{X}_t)Q(\mathbf{X}'_t;\mathbf{X}_t)}\right], \tag{9}$$

where $p(\mathbf{Y}_t|\mathbf{X}'_t)$ denotes the likelihood term over the state $\mathbf{X}'_t$ and $Q(\mathbf{X}'_t;\mathbf{X}_t)$ represents the proposal density.

Our algorithm combines the density of states term with the acceptance ratio in (9). Let M be a mapping function from the state $\mathbf{X}_t$ to the subregion S_i^p which contains the position state, $\mathbf{X}_t^p$ of $\mathbf{X}_t$.

$$M : \mathbf{X}_t \rightarrow \mathbf{S}_i^p. \tag{10}$$

Then the modified acceptance ratio is

$$a = min\left[1, \frac{p(\mathbf{Y}_t|\mathbf{X}'_t)^\alpha p^*(\mathbf{X}'_t)Q(\mathbf{X}_t;\mathbf{X}'_t)}{p(\mathbf{Y}_t|\mathbf{X}_t)^\alpha p^*(\mathbf{X}_t)Q(\mathbf{X}'_t;\mathbf{X}_t)}\right] = min\left[1, \frac{\frac{p(\mathbf{Y}_t|\mathbf{X}'_t)^\alpha}{g(M(\mathbf{X}'_t))}Q(\mathbf{X}_t;\mathbf{X}'_t)}{\frac{p(\mathbf{Y}_t|\mathbf{X}_t)^\alpha}{g(M(\mathbf{X}_t))}Q(\mathbf{X}'_t;\mathbf{X}_t)}\right], \tag{11}$$

where $p^*(\mathbf{X}'_t)$ denotes the approximated prior term in (7), $g(M(\mathbf{X}'_t))$ expresses the density of states at the subregion that includes the position state $\mathbf{X}_t^{p'}$ of $\mathbf{X}'_t$, and α indicates the weighting parameter. Our acceptance ratio in (11) has two different properties compared to that in (9). The first property is that (11) provides a way to escape from local maxima and reach to the global maximum. This property is crucial to the success of our tracking algorithm. If an abrupt motion exists in a scene, the algorithm has to sample the states in larger areas to deal with that motion where the Markov Chain has higher chances of meeting local maxima. In our acceptance step, the Markov Chain is guided by the ratio between the likelihood score and the density of states score, $\frac{p(\mathbf{Y}_t|\mathbf{X}_t)}{g(M(\mathbf{X}_t))}$. At a local maximum, this ratio initially has a large value since the likelihood has the higher score around the local maximum. However, while the simulation goes on, the ratio continues to decrease as the density of states generally increases around

(a) Escaping process (b) Tracking results (c) Density of states

Fig. 3. Properties of our acceptance ratio. (a) If the density of states in region 3 is much larger than one in region 4, the proposed state can be accepted although the state has a lower likelihood score than that of current state. (c) The brighter the color, the larger the density of states. Our method gets more samples at regions of two boxers while exploring all subregions at least to some degree.

local maxima. The proposed state is accepted when the ratio over the current state sufficiently decreases compared to one over the proposed state. Figure 3(a) illustrates the process of escaping from the local maximum.

As the second property, (11) efficiently schedules a sampling procedure so that the Markov Chain resides in a local maximum for a longer period, while guaranteeing to explore the whole state space at least to some degree. This property provides increased flexibility over the proposal density in (8). Note that the density of states term allows chances for the proposed state to be accepted at rarely visited subregions. On the other hand, the likelihood term forces the proposed state to be frequently accepted around local maxima. Since the length of the Markov Chain is limited, these two terms form the trade-off relationship. α in (11) controls this trade-off relationship. Higher weights on the likelihood term result in increasing the accuracy of MAP estimate. Conversely, the density of states term has to be increasingly weighted to cover the whole state space. Our acceptance ratio efficiently deals with this trade-off relationship as shown in Figure 3(c).

4.5 Wang-Landau Recursion Step

In this section, we propose a new efficient step called the Wang-Landau recursion to calculate the density of states $g(M(\mathbf{X}_t))$ in (11). This step follows the similar procedure as in the original Wang -Landau algorithm discussed in Section 3. Figure 4 provides the detailed process of our WLMC based tracking method that include the proposal, acceptance and Wang-Landau recursion step. This figure shows how the density of states is adapted for our tracking problem.

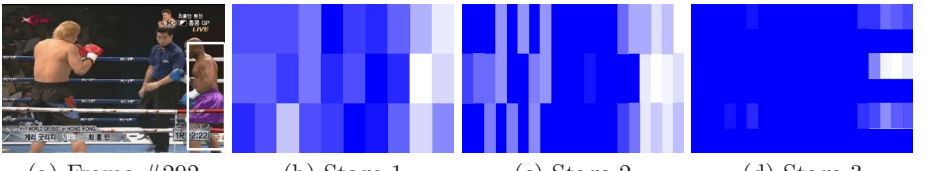
The key point is that the Wang-Landau recursion step addresses the chicken-egg-type problem. In order to estimate the density of states accurately, the acceptance ratio in (11) should guide the Markov Chain in the direction of producing a flat histogram over the whole state space. While, to calculate the acceptance ratio in (11), the density of states has to be known in advance. This recursion step provides a systematic way to visit all the subregions at least to some degree and

1) Initialize the Wang-Landau recursion:
Given d disjoint subregions S_i^p,
$g(S_i^p)=1,\ h(S_i^p)=0 \quad for\ i=1,\ldots,d$ and set $f=f_0=2.7$.

2) MCMC sampling: Repeat N times, where N is the total number of samples
a) Given the current state X_t^n (n-th sample at time t),
propose the new state X_t' using proposal density (8).
b) Compute the acceptance ratio (11),
If accepted, $X_t^{n+1}=X_t'$, other wise, $X_t^{n+1}=X_t^n$.
c) Wang-Landau update:
Update $g(M(X_t^{n+1}))\leftarrow g(M(X_t^{n+1}))*f \ ,h(M(X_t^{n+1}))\leftarrow h(M(X_t^{n+1}))+1$.
If the histogram is flat, reinitialize $h(S_i^p)=0$ for all i and set $f\leftarrow\sqrt{f}$.

3) Compute the MAP estimate X_t^{MAP}.

Fig. 4. WLMC based tracking method



(a) Frame #292 (b) Stage 1 (c) Stage 2 (d) Stage 3

Fig. 5. Annealing process. A-WLMC sequentially reduces $\mathbf{S}^p$ from (b) to (d) using the density of states. Then, A-WLMC leaves some subregions that contain robust candidates of the target position and eventually tracks the target as shown in (a).

simultaneously acquire the exact density of states. Note that the Wang-Landau algorithm typically converges although it does not satisfy detailed balance [15].

5 Annealed WLMC Based Tracking Method

We extend our WLMC based tracking method to an annealed version (A-WLMC) for more efficient sampling. In A-WLMC, the Markov Chain is defined over the annealed state space. And A-WLMC concentrates sampling on theses annealed subregions that compactly contain the target object. Figure 5 shows the process on how the state space is reduced to the interesting subregions. The algorithm starts the process over the whole state space and performs each stage using the WLMC based tracking method presented in Section 4. At the end of each stage, the state space is reduced by half, and the Chain is restarted over the reduced state space.

We utilize the density of states to anneal the state space. The state space basically consists of d disjoint subregions. Since the density of states becomes larger

1) Process the WLMC based tracking method.

2) Annealing step: if the histogram is flat,

a) Choose $d/2$ number of subregions S_i^p according to $g(S_i^p)$ in descending order and represent the chosen subregions as S_i^c *for* $i = 1, \ldots d/2$.

b) Divide each S_i^c into two regions: $S_{i_1}^c$ and $S_{i_2}^c$.

c) Update the density of states and subregions.

$$S_{2i-1}^a = S_{i_1}^c, \; S_{2i}^a = S_{i_2}^c \qquad \textit{for} \quad i = 1, \ldots d/2.$$

$$g(S_{2i-1}^a) = g(S_{2i}^a) = g(S_i^c) \qquad \textit{for} \quad i = 1, \ldots d/2.$$

$$S_i^p = S_i^a \qquad \textit{for} \quad i = 1, \ldots d.$$

where S_i^a represents the annealed subregion.

Fig. 6. Annealed WLMC based tracking method

around the local maxima, we choose the $d/2$ number of subregions according to the density of states in descending order. The chosen subregions are individually divided into two regions so that the total number of subregions becomes d again. The overall procedure of the annealed version is summarized in Figure 6.

6 Experimental Result

In this paper, the observation model utilized the HSV color histogram as a feature which is known to be robust to the illumination changes, and Bhattacharyya coefficient as a similarity measure [16]. We tested three video sequences: Seq.1, Seq.2 for camera shot changes, and; Seq.3 for rapid motions.[1] For the fair comparison, we used equal number of samples; 600, and compared the proposed algorithm with five different tracking methods: standard MCMC is based on [4]. Proposal variances are separately set to 8, 4 for the x, y direction; Adaptive MCMC is based on [17]; Quasi-random sampling is based on [13]; Particle filter is based on [3]. The motion model utilized the second-order autoregressive process and noise model is defined by the gaussian function of which the variance is set to 250; Mean shift is based on the implemented function in opencv.

6.1 Quantitative Comparison

Coverage test: The recall ρ and the precision ν measure the configuration errors between the ground truth state and the estimated state.

$$\rho = \frac{A_t^X \cap A_t^G}{A_t^G}, \quad \nu = \frac{A_t^X \cap A_t^G}{A_t^X}, \tag{12}$$

[1] The tracking results are available at $http://cv.snu.ac.kr/WLMC\_tracking.html$.

Table 1. ***F-measure*** in Seq. 1. As α value increases, the weight on the likelihood term in (11) also increases.

α	A-WLMC	Adaptive MCMC	Standard MCMC	Quasi-random	Particle filter
0.5	0.783126	0.780664	0.773494	0.726511	0.715354
1.0	0.795748	0.774263	0.769159	0.726511	0.715354
1.5	0.816221	0.773265	0.756279	0.726511	0.715354

where A_t^X denotes the estimated area and A_t^G indicates the ground truth area at time t. For good tracking quality, both the recall and precision should have high values. In information retrieval literatures, *F-measure* is often used for evaluating this quantity.

$$F = \frac{2\nu\rho}{\nu + \rho}. \tag{13}$$

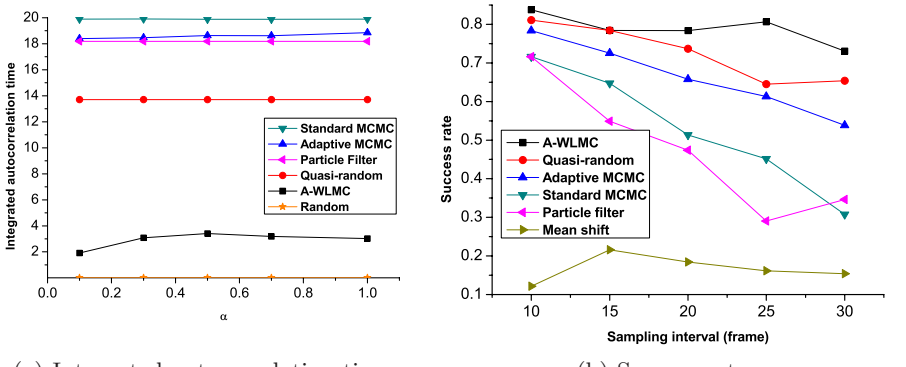
When the ground and estimated area perfectly overlap, *F-measure* is 1.0.

We obtained the ground truth states by manually drawing the bounding box around the target. Note that, in other methods, the state is re-initialized to the ground truth before they fail to track the abrupt motion. Then, the results in Table 1 indicates the accuracy of tracking the *smooth* motion, and states that the A-WLMC method is also as good as existing methods in the smooth motion case.

Autocorrelation time: The autocorrelation time measures the degree of statistical independence between samples. This independence property is important in terms of sampling efficiency. If samples are highly correlated, the statistical error does not decrease at the rate of the square root of the number of samples.

Let us define the autocorrelation function as follows:

$$C_{xx}(k) = \mathbb{E}\left[(\mathbf{X}_t^n - \mathbb{E}[\mathbf{X}_t^n])(\mathbf{X}_t^{n+k} - \mathbb{E}[\mathbf{X}_t^{n+k}])\right], \tag{14}$$



(a) Integrated autocorrelation time (b) Success rate

Fig. 7. Evaluation of the tracking methods. (a) Integrated autocorrelation time at Seq. 2. (b) Success rate at Seq. 3 as a function of down-sampling interval for different tracking methods.

where $\mathbb{E}$ is the expectation operator, $\mathbf{X}_t^n$ and $\mathbf{X}_t^{n+k}$ are the n-th and $(n+k)$-th samples at time t , respectively. This function generally decays exponentially by the number of samples k such that,

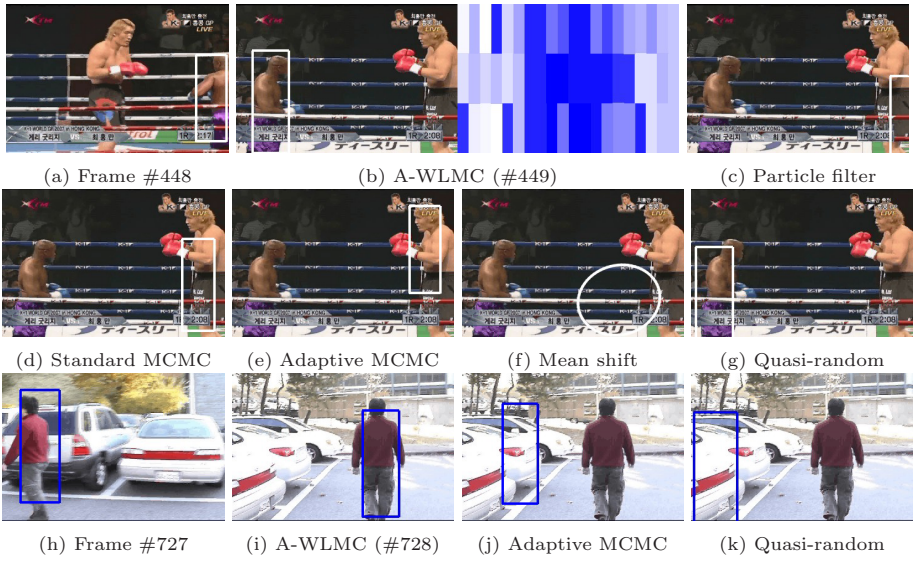
$$C_{xx}(k) \approx \exp(-\frac{k}{\tau_{exp}}), \tag{15}$$

where τ_{exp} is the *exponential autocorrelation time*. For the computational simplicity, we use *integrated autocorrelation time* τ_{int} suggested by [18].

$$\int_0^\infty C_{xx}(k)dt = \int_0^\infty C_{xx}(0)\exp(-\frac{k}{\tau_{int}})dk = \tau_{int}C_{xx}(0), \tau_{int} = \sum_k \frac{C_{xx}(k)}{C_{xx}(0)}. \tag{16}$$

Figure 7(a) displays the efficiency of the A-WLMC method to produce both uncorrelated and meaningful samples. Although the random sampling method is the best in terms of statistical independence, this method is not guided in a principled manner. In contrast, A-WLMC is the winner in terms of efficiency. A-WLMC guarantees samples to converge to the target density and simultaneously generates higher uncorrelated samples than those by other methods.

Success rate: If *F-measure* is larger than 0.5, the target is considered as correctly tracked at that frame. The success rate indicates the ratio between the number of correctly tracked frames and the number of total frames. For this test we down-sampled Seq. 3 with the sampling interval from 10 frames to 30 frames. The results are depicted in Figure 7(b). In comparison with the other results, the success rate of the A-WLMC method is less affected by the change of the sampling interval, whereas those of other methods rapidly decrease as the sampling

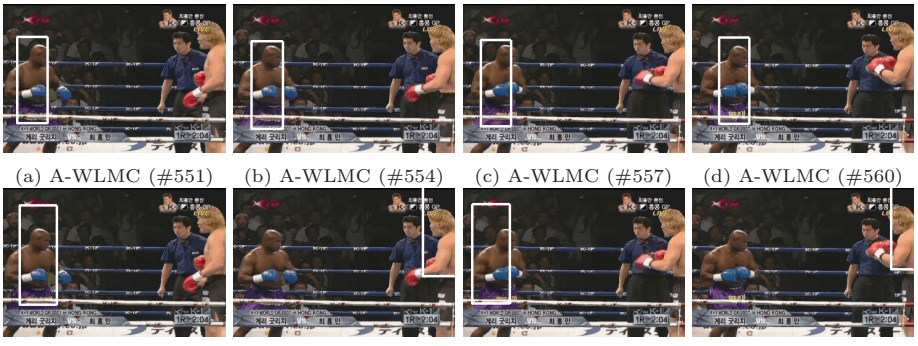


(a) Frame #448 (b) A-WLMC (#449) (c) Particle filter

(d) Standard MCMC (e) Adaptive MCMC (f) Mean shift (g) Quasi-random

(h) Frame #727 (i) A-WLMC (#728) (j) Adaptive MCMC (k) Quasi-random

Fig. 8. Tracking results when the camera shot change occurs in Seq. 2, Seq. 1



(a) A-WLMC (#551) (b) A-WLMC (#554) (c) A-WLMC (#557) (d) A-WLMC (#560)

(e) Quasi-rand (#551) (f) Quasi-rand (#554) (g) Quasi-rand (#557) (h) Quasi-rand (#560)

Fig. 9. Tracking results of the smooth motion case in Seq. 2

(a) A-WLMC (#18) (b) A-WLMC (#22) (c) A-WLMC (#25)

Fig. 10. Tracking results in videos where rapid motions exist. In Seq. 3, the video is down-sampled to 25 sampling interval.

interval increases. It is significant to note that A-WLMC successfully tracks the target even in highly down-sampled video which contains severe abrupt motions.

Speed: A-WLMC has no additive computational burden compared to the other sampling based tracking methods since the density of states can be calculated at extremely less computational cost. A-WLMC runs at 1∼10 fps for 320×240 videos. Note that our code is not optimized.

6.2 Qualitative Comparison

Figure 8 presents the tracking results in the camera shot change case. In the video, A-WLMC successfully tracked the target whereas other methods failed to escape from the previous position of the target. The quasi-random method also tracked the abrupt motion in Seq. 2 (Figure 8(g)) whereas the method failed to track the motion in Seq. 1 (Figure 8(k)). We also illustrate the density of states obtained at frame #449 of Seq. 2 in the right part of Figure 8(b). The whiter regions indicate that A-WLMC got more samples at those regions which can be regarded as local maxima. As shown in the figure, there are a number of local minima found by A-WLMC. This means that our method has a ability to escape one local maximum and reach to another one. Furthermore, we

tested to track the target of which motion is smooth. As A-WLMC and quasi-random sampled states at the larger portions of the state space compared with the conventional tracking methods, it is very crucial to check the accuracy of tracking the smooth motion and robustness to the clutters. As shown in Figure 9, A-WLMC accurately tracked the target of which motion is smooth over time. In contrast, the quasi-random sampling was easily distracted by clutter and often got trapped in local maxima which are the right boxer at the video. Figure 8 and 9 demonstrate that A-WLMC well tracks the smooth and abrupt motion at the same time compared with the other tracking methods. Note that most of the tracking performance comes from the A-WLMC *filter* rather than randomness of the proposal density. Quasi-random also used the similar proposal density, but which results were worse. As another example of an abrupt motion, we tested down-sampled video that included rapid motions of which directions and distances were quite unpredictable. A-WLMC addressed this uncertainty of the motion and accurately proposed the object's position as shown in Figure 10.

7 Conclusion

In this paper, we have proposed an effective tracking algorithm based on the Wang-Landau Monte Carlo. The algorithm efficiently addresses tracking of abrupt motions while smooth motions are also accurately tracked. Experimental results demonstrated that the proposed algorithm outperformed conventional tracking algorithms in severe tracking environments. Our current algorithm has not fully considered the abrupt changes in appearance. We leave this problem to be addressed in future research studies.

Acknowledgement

This research was supported in part by the Defense Acquisition Program Administration and Agency for Defense Development, Korea, through the Image Information Research Center under the contract UD070007AD, and in part by the MKE (Ministry of Knowledge Economy), Korea under the ITRC (Information Technolgy Research Center) Support program supervised by the IITA (Institute of Information Technology Advancement) (IITA-2008-C1090-0801-0018)

References

1. Yilmaz, A., Javed, O., Shah, M.: Object tracking: A survey. ACM Comput. Surv. 38(4) (2006)
2. Cai, Y., de Freitas, N., Little, J.: Robust visual tracking for multiple targets. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3954, pp. 107–118. Springer, Heidelberg (2006)
3. Isard, M., Blake, A.: Icondensation: Unifying low-level and high-level tracking in a stochastic framework. In: Burkhardt, H., Neumann, B. (eds.) ECCV 1998. LNCS, vol. 1407. Springer, Heidelberg (1998)

4. Khan, Z., Balch, T., Dellaert, F.: An mcmc-based particle filter for tracking multiple interacting targets. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3024, pp. 279–290. Springer, Heidelberg (2004)
5. Smith, K., Perez, D.G., Odobez, J.: Using particles to track varying numbers of interacting people. In: CVPR (2005)
6. Zhao, T., Nevatia, R.: Tracking multiple humans in crowded environment. In: CVPR (2004)
7. Wu, B., Nevatia, R.: Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors. IJCV 75(2), 247–266 (2007)
8. Nillius, P., Sullivan, J., Carlsson, S.: Multi-target tracking: Linking identities using bayesian network inference. In: CVPR (2006)
9. Jepson, A., Fleet, D., Maraghi, T.E.: Robust online appearance models for visual tracking. PAMI 25(10), 1296–1311 (2003)
10. Yang, M., Wu, Y.: Tracking non-stationary appearances and dynamic feature selection. In: CVPR (2005)
11. Han, B., Davis, L.: On-line density-based appearance modeling for object tracking. In: ICCV (2005)
12. Wang, F., Landau, D.: Efficient, multiple-range random walk algorithm to calculate the density of states. Phys. Rev. Lett. 86(10), 2050–2053 (2001)
13. Philomin, V., Duraiswami, R., Davis, L.: Quasi-Random Sampling for Condensation. In: Vernon, D. (ed.) ECCV 2000. LNCS, vol. 1843, pp. 134–149. Springer, Heidelberg (2000)
14. Li, Y., Ai, H., Yamashita, T., Lao, S., Kawade, M.: Tracking in low frame rate video: A cascade particle filter with discriminative observers of different lifespans. In: CVPR (2007)
15. Atchade, Y., Liu, J.: The wang-landau algorithm for monte carlo computation in general state spaces. Technical Report (2004)
16. Perez, P., Hue, C., Vermaak, J., Gangnet, M.: Color-based probabilistic tracking. In: Tistarelli, M., Bigun, J., Jain, A.K. (eds.) ECCV 2002. LNCS, vol. 2359, pp. 661–675. Springer, Heidelberg (2002)
17. Roberts, G., Rosenthal, J.: Examples of adaptive mcmc (preprint, 2006)
18. Berg, B.: Introduction to markov chain monte carlo simulations and their statistical analysis. Phys.Stat. Mech. (2004)