Object Tracking in Frame-skipping Video Acquired Using Wireless Consumer Cameras

Regular Paper

Anlong Ming¹,*, Huadong Ma¹ and Charles X. Ling¹

1 Senior Member, IEEE
* Corresponding author E-mail: minganlong@bupt.edu.cn

Received 26 Apr 2012; Accepted 16 Jul 2012

DOI: 10.5772/51613

© 2012 Ming et al.; licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract Object tracking is an important and fundamental task in computer vision and its high-level applications, e.g., intelligent surveillance, motion-based recognition, video indexing, traffic monitoring and vehicle navigation. However, the recent widespread use of wireless consumer cameras often produces low quality videos with frame-skipping and this makes object tracking difficult. Previous tracking methods, for example, generally depend heavily on object appearance or motion continuity and cannot be directly applied to frame-skipping videos. In this paper, we propose an improved particle filter for object tracking to overcome the frame-skipping difficulties. The novelty of our particle filter lies in using the detection result of erratic motion to ameliorate the transition model for a better trial distribution. Experimental results show that the proposed approach improves the tracking accuracy in comparison with the state-of-the-art methods, even when both the object and the consumer are in motion.

Keywords Frame dropping, Low frame rate, Frame-Skipping, Particle filter, Wireless consumer camera, Tracking

1. Introduction

Object tracking, in general, is the tracking of an object or objects over a sequence of images. Object tracking is an important task in the field of computer vision and it is usually performed at an early stage in the context of higher-level applications such as automated surveillance, motion-based recognition, video indexing, traffic monitoring and vehicle navigation [1]-[3]. For example, in an automated surveillance system there are at least three key steps: detection of interesting objects, tracking of such objects over frames and analysis of object trajectories to recognize their behaviour. Therefore, object tracking is a critical task in many high-level applications.

Object tracking is a very challenging problem because a lot of difficulties can arise due to non-rigid object structures, occlusions, changing appearance patterns of both the object and the scene, etc. There have been many methods designed to overcome these common difficulties. However, the availability of low-cost hardware, such as CMOS cameras and microphones that are able to ubiquitously capture video content from the
environment, has fostered the development of wireless video sensor networks (WVSNs) [4, 5]. Wireless devices (Fig. 1 shows the wireless consumer cameras used in our experiment in Section 5) allow retrieving videos and tracking in WVSNs is a practical requirement of many real-time applications. The retrieved videos, however, usually have two common difficulties which are usually named together as a frame-skipping problem (see Fig. 2): one is unexpected frame dropping (missing frames in a continuous video sequence) and the other is low frame rate. The frame-skipping problem can be caused by various factors, e.g., low hardware cost, low or unstable processing speed in the video sources, frame dropping caused by the transmission conditions or online compressing or uncompressing which limits the frame rate. Therefore, videos with frame-skipping is common and “normal” for WVSNs and the property of the video flow itself - frame-skipping has become an important issue in many applications.

Previous tracking methods, in general, depend heavily on object appearance or motion continuity (see Section 2 for a detailed review). These methods often utilize the assumption of temporal continuity, whereas in frame-skipping videos the continuity of a target is often too weak to follow. Essentially, frame-skipping videos create difficulties in obtaining the transition model (describing how objects move between frames). Meanwhile, identifying the target from frame to frame is difficult due to the absence of context and we cannot rely only on image processing techniques. Therefore, most previous tracking methods cannot be directly applied to frame-skipping videos. One feasible solution proposed previously, whether or not the motive is frame-skipping tracking, is the integration of object detection and tracking because object detection allows for discrimination of the target from the others [6]-[14]. The solution can overcome the frame-skipping difficulties partially, but applying reliable object detection over a large search space is often costly [6]-[12]. Furthermore, identifying the target requires strong discriminative power which is usually achieved by massive offline training [13], [14]. However, in many applications of WVSNs, offline training is impossible because both targets and scenarios are unpredictable. Our method, on the other hand, is highly efficient in frame-skipping videos and it does not require offline training.

Another important challenge of object tracking is that consumer cameras are frequently rotated or moved during the video capturing process. The motion of consumer cameras also makes object tracking more difficult because the non-stationary background is an obstacle to the extraction of moving objects, thus static background subtraction-based methods [6]-[10] are not applicable. When camera motion and possible cluttered backgrounds appear in some applications, a particle filter (using a dynamic model to guide the particle propagation within a limited sub-space of target state) has been used previously to solve object tracking effectively [15], [21]. However, when object motion becomes unpredictable under frame-skipping conditions, the standard particle filter will cause departure of the sample set from the true target state and this eventually leads to tracking loss.

Figures 1 and 2. (a) Normal video frame sequences; (b) Frame dropping; (c) Low frame rate.
In this paper, we propose an improved particle filter with a better transition model to overcome the difficulties of object tracking in frame-skipping videos. In our method, it is motion detection rather than object detection as in [6]-[14] that plays the key role in object tracking. We apply fast and reliable detection which produces a global description in an acceptable search space. The novelty of our particle filter lies in using the detection result of erratic motion to ameliorate the transition model for a better trial distribution. We compare the tracking accuracy of the proposed approach with the state-of-the-art methods and show that our new method is much better, even when both the object and the consumer camera are in motion.

The remainder of this paper is organized as follows: Section 2 briefly summarizes related works on frame-skipping videos. Section 3 is devoted to analysing the essence of the frame-skipping problem in the probabilistic framework and then reveals the deficiency of the standard particle filter. Section 4 introduces the extraction of erratic motion and then proposes our particle filter with a newly defined transition model for a better trial distribution. Section 5 presents the experimental results which show that our new method outperforms the state-of-the-art methods and we also discuss the limitations of our method. Section 6 concludes this paper.

2. Related Works

The frame-skipping events, in general, are equivalent to uncertain erratic motion in most cases. A large number of the state-of-the-art methods, such as mean shift [18], [19], generally require the kernels or feature patches in consecutive frames to overlap with or be in a very close vicinity of each other. However, some existing publications [6]-[14] have attempted to tackle similar difficulties, whether or not the motive is partial frame-skipping tracking. A common feature of all these methods is the integration of object detection and tracking. Furthermore, we classify these works into three categories.

i. “Global object detection” for object tracking. These methods use an independent detector to guide the search of an existing tracker when target motion becomes unpredictable and require an object detector fast enough to be applied to the whole frame in most cases. Okuma et al. [13] use a boosted detector to amend the trial distribution of the particle filter. However, the boosted detector requires massive offline training. Another similar piece of research on mixture trial distribution is described in [6]. Porikli et al. [17] extend the standard mean shift technique using multiple kernels at motion areas detected by background subtraction to track in both 6 fps (frame per second) and 1 fps camera fixed videos. In our method, we utilize erratic motion detection (this requires no offline learning) to conquer the frame-skipping problem.

ii. “Object detection and connection” for object tracking. These methods detect the objects of interest and then constructing trajectories by analysis of motion continuity, object appearance similarity, etc. However, the algorithms of this category [8]-[10] are limited in static background scenes, where a fast change detector is easily to be realized. Besides, the trajectories are uncertain and usually cannot be recognized in frame-skipping videos. The methods in ii are not applicable in many applications of WVSNs because of non-stationary backgrounds and require an object detector fast enough to be applied to the whole frame in most cases. In our method, the improved particle filter can be applied to dynamic background scenes.

iii. “Multi-scale or multi-stage object detection” for object tracking. These methods increase the discriminative power by layered sampling of multi-scale likelihoods or multi-stage observations. In [11], multi-scale approaches are designed for erratic motion by layered sampling of multi-scale likelihoods [12]. However, the multi-scale approaches adopt the same observation model but lose image information in down-scaling process. Li et al. [14] propose a cascade particle filter with discriminative observers of different life spans. This method can be viewed as a classification problem in the sense of distinguishing tracking human face from the background. Besides, in the long span of this method, massive offline training costs several days. In our method, we integrate erratic motion detection and tracking together to find out a way in getting a better transition model (rather than others, e.g., the observation model proposed in [14]) to conquer the frame-skipping problem without massive offline learning.

3. Problem Analysis

The standard particle filter can effectively overcome the difficulties such as camera motion and clustered backgrounds which usually appear in applications of WVSNs. However, when object motion becomes erratic and unpredictable under frame-skipping conditions, the standard particle filter makes for tracking loss. Hereby we first briefly review the basic rules of object tracking in a probabilistic frame work and then reveal the deficiency of the standard particle filter in case of frame-skipping.

3.1 Tracking in a Probabilistic Framework

The basic idea of particle filter, which means tracking in a probabilistic framework, is now briefly reviewed here firstly.
To define the tracking problem, we can consider a dynamic system represented by the stochastic process \( \{x_k, k \in \mathbb{N}\} \) of a target given by
\[
x_k = f_k(x_{k-1}, v_{k-1})
\]
(1)
where \( f_k : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n \) is a possibly nonlinear function of the state \( x_{k-1}; \{x_{k-1}, k \in \mathbb{N}\} \) is an i.i.d. process noise sequence; \( n_x, n_v \) are dimensions of the state and process noise vectors, respectively and \( x_k \) is the hidden state such as object scale, location.

Then tracking problem can be converted to recursively estimate \( x_k \) from observations
\[
z_k = h_k(x_k, n_k)
\]
(2)
where \( h_k : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n \) is a possibly nonlinear function; \( \{n_k, k \in \mathbb{N}\} \) is an i.i.d. observation noise sequence; \( n_x, n_v \) are dimensions of the observation and observation noise vectors, respectively, and \( z_{k,i} \) is the set of all available observations up to time \( k \).

Given the data \( z_{k,i} \), from a Bayesian perspective, we seek recursively calculation of the probability distribution function (pdf) \( p(x_k \mid z_{k,i}) \) which performs forward inference using the Bayesian filtering distribution:
\[
p(x_k \mid z_{k,i}) \propto p(z_k \mid x_k) \int p(x_k \mid x_{k-1}) p(x_{k-1} \mid z_{k-1}) dx_{k-1}
\]
(4)

Where the normalizing constant \( \alpha \) depends on the likelihood function defined by the observation model in (2) and the known statistics of \( n_k \).
\[
\alpha = 1 / p(z_k \mid x_k) = 1 / \int p(z_k \mid x_k) p(z_{k,i}) dx_k
\]
(5)

Note that in (3):
- \( p(x_k \mid x_{k-1}) = p(x_k \mid x_{k-1}, z_{k,i}) \) is made as (1) describes a Markov process of order one [20].
- The observation \( z_k \) is used to modify the prior density and obtain the required posterior density of the current state.

Equation (3) describes the optimal Bayesian solution. However, this recursive propagation of the posterior density is only a conceptual solution and it cannot be determined analytically. So we need a method to approximate the optimal Bayesian solution such as a particle filter.

A particle filter [15], [21] uses a probabilistic framework to formulate tracking as an inference in a Hidden Markov Model (HMM). It is based on random measurement density approximated by a set of weighted particles. Each particle consists of the state domains and its corresponding probability (weight) is denoted by \( \{x_{k,i}, w_k^{\gamma_i} \} \), where \( i \) is the particle number and \( N_k \) is the number of particles. The weights are normalized such that \( \sum w_k^{\gamma_i} = 1 \). Then, the posterior density at time \( k \) can be approximated as
\[
p(x_k \mid z_{k,i}) \approx \sum_{i=1}^{N_k} w_k^{\gamma_i} \delta(x_{k,i} - x_k)
\]
(6)
where \( \delta() \) is the Dirac delta measure. Let \( x^* \sim q(x) \), \((i = 1, ..., N_k)\) be samples that are generated from a proposal called an importance density. If the samples were drawn from an importance density \( q(x_{k,i} \mid z_{k,i}) \) then the weights in (5) are defined to be
\[
w_k^{\gamma_i} \propto p(z_{k,i} \mid x_{k,i}) \propto q(z_{k,i} \mid x_{k-1,i}) p(x_{k-1,i} \mid z_{k-1,i})
\]
\[
\propto w_{k-1}^{\gamma_i} p(z_k \mid x_{k-1,i}) p(x_k \mid x_{k-1,i}) p(x_{k-1,i} \mid z_{k-1,i})
\]
\[
\propto w_k^{\gamma_i} p(z_k \mid x_{k-1,i}) p(x_k \mid x_{k-1,i})
\]
(7)

**Figure 3.** Examples of HSV histogram-based observation model.
3.2 Deficiency of the Standard Particle Filter

In the standard particle filter, the distribution $p(x_k | z_{1:k})$ may be obtained, recursively, in two well-known stages: prediction and update.

- **Prediction:**
  
  $$p(x_k | z_{1:k-1}) = \int p(x_k | x_{k-1}) p(x_{k-1} | z_{1:k-1}) dx_{k-1}.$$  

- **Update:** $p(x_k | z_{1:k})$ illustrated in (3).

According to the standard particle filter, the calculation of the integral in (8) is carried out by importance sampling, which means samples that are generated from a proposal distribution.

In practice, a presumed prior distribution $p(x_k | x_{k-1})$ is widely used as the proposal distribution. But when object motion becomes erratic and unpredictable under frame-skipping conditions, such transition model $p(x_k | x_{k-1})$ will cause departure of the sample set from the true target state which eventually leads to tracking loss. Hereby we give an implement case of the standard particle filter to show the tracking loss in a frame-skipping video. In our case:

- Prior distribution is simply an impulse based on user input. It describes initial distribution of object states and could be based on an object detector.

- Observation model is a simple HSV histogram-based model (Fig. 3). We specify the likelihood of an object being in a specific state. Likelihood is based on a distance metric $D[;]$ between histograms $h_0$ (or $h(0)$) (the original one) and $h(x_k)$. Then the observation model is denoted as

  $$p(z_k | x_k) \propto e^{-D(h_0, h(x_k))}$$  

(9)

Transition model is a Gaussian window around current state where the standard particle filter usually samples the next state from. As to a given particle at time $k-1$, the prediction and measurement process is shown in Fig. 4.

---

**4. Our Improved Particle Filter**

Erratic motion detection can produce a global description in a search space. So in Section A, we present how to represent and extract erratic motion as a local motion vector. Then in Section B, the transition model is redefined in the view of target drift over frames based on the local motion vector. Finally in Section C, we describe how this new transition model is integrated into the probabilistic framework to tackle the frame-skipping problem.

**4.1 Representation and Extraction of Erratic Motion**

In this paper, we use MHI and its hierarchical mechanism [16], [17], a simple and fast enough method to represent motion in successively layered silhouettes which directly encode system time. This representation can be used to segment and measure the motions induced by the object in a video scene. These segmented regions are not "motion blobs", but motion regions naturally connected to the moving parts of the object of interest. First, we label those pixels (a set number of standard deviations from the mean RGB background) as foreground. Then a pixel dilation and region growing method is applied to remove noise and extract the silhouette. Then MHI representation is constructed by successively layering selected image regions over time using a simple update rule:

$$\text{MHI}_j(x, y) = \begin{cases} 
\tau, & \text{if } \Phi(l(x, y)) \neq 0 \\
0, & \text{else if } \text{MHI}_j(x, y) < (\tau - \theta) 
\end{cases}$$  

(10)

where each pixel $(x, y)$ in the MHI is marked with a current timestamp $\tau$ if the function $\Phi$ indicates object...
(or motion) presence in the current video frame $I(x, y)$; the remaining timestamps are removed if they are longer than the decay value $(\tau - \theta)$.

![Figure 5. Tracking in a frame-skipping video lab.avi at 3-4 fps. (a) Using a standard particle filter (particle number = 400) begins at #2 and completely fails at #17. (b) Using our method (particle number = 50), successful tracking (#13 to #22).](image)

Since motion can be perceived from the displayed timestamp gradients in the template, we could convolve gradient masks with the timestamp values in the MHI to extract a motion vector at each pixel. Gradients of the MHI can be calculated efficiently by convolution with separable Sobel filters in the X and Y directions yielding the spatial derivatives: $F_x(x, y)$ and $F_y(x, y)$.

A simple calculation for the global weighted orientation is denoted as (11), where $\phi$ is the global motion orientation; $\phi_{ref}$ is the base reference angle (peaked value in the histogram of orientations); $\text{norm}(r, \theta, \text{MHI}_s(x, y))$ is a normalized MHI value; $\text{angDiff}(\phi(x, y), \phi_{ref})$ is the minimum, signed angular difference of an orientation from the reference angle; gradient orientation at each pixel is denoted as: $\phi(x, y) = \arctan \left( \frac{F_y(x, y)}{F_x(x, y)} \right)$.

$$\phi = \phi_{ref} + \frac{\sum_{r,\theta} \text{angDiff}(\phi(x, y), \phi_{ref}) \times \text{norm}(r, \theta, \text{MHI}_s(x, y))}{\sum_{r,\theta} \text{norm}(r, \theta, \text{MHI}_s(x, y))}$$  (11)

However, the use of a discrete fixed-sized gradient mask (i.e., $3 \times 3$ Sobel gradient masks) limits the range of recoverable motion. When an object moves at different velocities, a fixed-sized mask can result in detection failure. Therefore, we use a hierarchical MHI mechanism [16], which extends the original MHI representation into a hierarchical pyramid format, to appropriately extract the MHI motions of different velocities. An image pyramid is constructed by recursively low-pass filtering and subsampling an image until reaching the desired size of spatial reduction. This permits us to use fixed-sized gradient masks at each pyramid level to calculate motions of different speeds. To create the corresponding MHI pyramid, each level from the $\Phi$ pyramid is used to update a MHI of that particular resolution. Then the algorithm to segment motion regions is denoted as follows:

**Step 1:** we choose the pyramid level $L$ with the minimum acceptable temporal disparity (finest temporal resolution): $L = \arg \min_f (F_x^f(x, y)^2 + F_y^f(x, y)^2)$  (12)

At level $L$:
- **Step 2:** scan the MHI until we find a pixel with the current timestamp (most recent silhouette).
- **Step 3:** go around the boundary of the current silhouette region looking outside for recent unmarked motion history “steps”. When a suitable step is found, mark it with a downward floodfill. If the size of the fill is not big enough, zero out the area.
- **Step 4:** store the segmented motion mask that was found.
- **Step 5:** if the boundary “walk” has not circumnavigated the current silhouette, go to Step 3.
- **Step 6:** calculate the centre $\phi_{cen}$ of all segmented motion masks; calculate the global motion orientation $\bar{\phi}$ using (11).

Some samples (#8 to #13) of erratic motion detection are shown in Fig. 6. This method is simple and affords completely real-time performance. It is noteworthy that the local motion vector $(\phi_{cen}, \bar{\phi})$ at the time $k+1$ indicates the erratic motion at the time $k$.

![Figure 6. Erratic motion detection. Top row: the frames of a frame-skipping video. Bottom row: MHIs and the local motion vectors (the lines in the red circles indicate the motion orientations) corresponding to the frames in the top row.](image)

### 4.2 The Integration of Motion Detection and Tracking

As what we have analysed in Section 3, when target motion becomes erratic and unpredictable under the frame-skipping conditions, the standard particle filter is
not applicable, e.g., sampling the next state from a Gaussian window around the current state is no longer a good choice. Therefore, based on the local motion vector mentioned above, we redefine the transition model described in (1) as follows:

\[ x_t = f_t(x_{t-1}, v_{t-1}) + v_{\text{drift}} \]  

(13)

Given a state \( x_{k-1} \), if erratic motion is detected at time \( k \), then we can represent the possible drift \( v_{\text{drift}} \) between time \( k-1 \) and \( k \) as:

\[ v_{\text{drift}} = \text{Drift}(\phi_{\text{cen}}, \bar{\phi}, \text{Dist}(\phi_{\text{cen}}, x_{k-1})) \]  

(14)

where \( \text{Drift}(\cdot) \) describes the state \( x_{k-1} \) drifts a distance \( \text{Dist}(\phi_{\text{cen}}, x_{k-1}) \) (between \( \phi_{\text{cen}} \) and \( x_{k-1} \) ) from \( \phi_{\text{cen}} \) along the orientation \( \bar{\phi} \). It is noteworthy that we should restrain the search space around the state \( x_{k-1} \) by a predefined threshold to reduce the false alarm rate within the bounds of an acceptable one. Then the final improved particle filter is presented in Algorithm 1.

**Algorithm 1** The improved particle filter

**Input:** \([x_{k-1}, w_{k-1}^{N_x}], z_{k-1}, (\phi_{\text{cen}}, \bar{\phi})\) at time \( k \)

**Output:** \([x_k, w_k^{N_x}]\)

**Begin**

1. if \( k=0 \) then
2. - Initialization: \( x_k' \sim p(x_0 | z_0) = p(x_0) \)
   \( w_k' = 1/N_x \)
3. for \( i = 1 : N_x \) do
4. if (Erratic motion is detected) then
5. - Draw: \( x_k \sim f_i(x_k^{-1}, z_k) + v_{\text{drift}} \)
6. else
7. - Draw: \( x_k \sim g(x_k | \phi_{\text{cen}}, z_k) \), see (1)
8. - Assign \( w_i' = \frac{p(z_k | x_i') p(x_i' | x_{k-1}')}{q(x_i' | x_{k-1}', z_k)} \)
9. Calculate total weight: \( t = \sum_{i=1}^{N_x} w_i' \)
   Normalize: \( w_i'' = \frac{w_i'}{t} \)
10. Calculate \( N_{\text{eff}} = 1/\sum_{i=1}^{N_x} (w_i'')^2 \)
11. if \( N_{\text{eff}} < N_{\text{threshold}} \) then
12. - Resampling: \( x_k'' \sim x_i'', \text{ so that } \Pr(x_k'' = x_i'') = w_i'' \)
   \( w_i'' = \frac{1}{N_x} \)
13. End.

**Figure 7.** The flowchart of a new prediction and measurement process in our transition model

**5. Experimental Results**

In this section we demonstrate the benefits of using the proposed method. We present the experimental setup in Section A, then the metrics for our method evaluation is summarized in Section B. By following these metrics, we analyse the efficiency of the sampling process in our method in Section C and do some quantitative comparisons of object tracking in Section D. Finally, the discussion of our method is given in Section E.

Theoretically, a good implementation of the transition model should take into account previous states for velocity and acceleration information. In this paper, we use an erratic motion detected, second-order, autoregressive dynamical model to predict the next state based on the previous two plus the Gaussian noise. Let \( x_k \) be the coordinate of a sampled particle at the next time \( k \), then

\[ f_i : \kappa_1(x_{k-2} - \bar{x}) + \kappa_2(x_{k-1} - \bar{x}) + \kappa_3 g_k \]  

(15)

Where \( \bar{x} \) is the centre coordinate of particles; \( g_k \) is Gaussian noise. In our case, a new prediction and measurement process is shown in Fig. 7.
\( \kappa_1 = 1.0 \). Moreover, we abbreviate the standard particle filter as PF in the experiments. The numbers of particles adopted in different cases are 50-1500.

5.2 Metrics for Evaluation

To verify the effectiveness of the proposed method, we get a systematic objective evaluation chiefly via following metrics [14]: the effective sample size (ESS) [21] analysis in the sampling process, tracking error and computational cost with similar tracking performance, respectively.

- ESS analysis in the sampling process. To analyse sampling efficiency, we compute ESS of importance sampling. In the same set of tracking sessions, ESS measures the uniformity of the weights of the particles and is defined by \( \text{ESS} = 1/\sum N_s w_i^2 \), where the larger the ESS is, the more particles are concentrated in the neighbourhood of the object to be tracked and thus, the better the chances are of the algorithm responding to fast changes.

- Tracking error. We compare effectiveness of the tracking methods in two aspects, position error in pixel and in size. Position error in pixel means the drift of a target location measured by pixel, while position error in size illustrates the scale inaccuracy of a target in a tracking process.

- Computational cost with similar tracking performance. The cost of our tracker can be divided into two parts:
  - A cost shared by the prediction and update of the particle filter, mainly depends on \( N_s \), namely, the number of particles calculated for each observer.
  - A calculation of the local motion vector and some basic image processing involved in the calculation.

5.3 Efficiency of the Sampling Process

With comparison to PF (with different particle numbers), a quantitative analysis of sampling efficiency is done on a test sequence (CAVIAR.avi) from CAVIAR datasets (free and open). CAVIAR.avi has been down-sampled randomly to corresponding 4-6 fps. Fig. 8 (a) includes a quantitative analysis of sampling efficiency by the curve of ESS.

We can simply describe ESS as that \( N_s \) (ESS value) samples drawn from the target distribution can approximate \( N_s \) weighted samples. Therefore, the higher ESS is, the better the sampling efficiency achieved by the system. Obviously, increasing the particle numbers of PF can compensate for its inaccuracy in the prediction stage under the frame-skipping conditions. When the particle number of PF is enlarged to 250 (5 times as much as the number in our method), the performance becomes similar to the proposed method. According to Fig. 8 (a), ESS is roughly in proportion to the particle number. In other words, the ability to predict the state of the target is enhanced with the cost of drawing more particles for a larger search space. The ESS of our method is even higher than PF with 250 particles, because in the update stage our method enables the convergence of particles around high-likelihood regions.

![Figure 8](image.png)

Figure 8. (a) EES curve on CAVIAR.avi. (b) Position error curve on CAVIAR.avi. The green windows indicate the person positions of every current frame, while those in yellow denote the person positions of every previous frame.

5.4 Quantitative Comparison

To validate the sampling effectiveness of the proposed method, quantitative comparisons of tracking are done via tracking errors in two aspects.

- With comparison to PF with different numbers of particles, a quantitative comparison of position error in pixel is done on the test sequence (CAVIAR.avi). Fig. 8 (b) illustrates a quantitative analysis of sampling effectiveness: the error rate of our method is still low with the smallest particle numbers; the curve of tracking error also includes that enlarging the particle number of PF can compensate for its poor accuracy under frame-skipping conditions.
A comparison of position error in size is done among PF, the mean shift using a colour histogram [18] and our method on test sequences: CAVIAR.avi, lab.avi and football.avi. lab.avi (Fig. 5) is taken by a wireless consumer camera just outside of our lab (3-4 fps). football.avi (Fig. 9 (a)) records a player engaged on a pitch (2-4 fps). All videos are down-sampled randomly to the corresponding frame rate. By the curve indicated in Fig. 10, our method shows higher accuracy than the others under the frame-skipping condition.

In addition, in our tracker: 1) the cost in the prediction and update stages of the particle filter mainly depends on \( N \). 2) For the calculation of the local motion vector, a hierarchical MHI mechanism is adopted to handle different velocities of moving targets efficiently and a pyramid of images is built so that, in each localized search space, an image is recursively low-pass filtered and sub-sampled until reaching the desired size of spatial reduction. Our experimental results show that the calculation of the local motion vector and the basic image processing involved is roughly comparable to PF with 50 particles.

Therefore, a comparison of the computational cost is done with the representation of average number of particles calculated by each observer per frame on three test sequences: CAVIAR.avi, lab.avi and football.avi. The result is shown in Fig. 11. The compared PF with different particles have similar tracking performance to our approach, yet many more excessive particles are calculated.

5.5 Discussion

Fig. 9 (a) shows a player engaged on a pitch. Challenges involved in this video sequence are common in real-life cases, e.g., the camera is moving and shaking when following the player’s movements, the player’s motion itself is unstable, with zoom in and zoom out, and the pose changes when he stands up to kick. Our method tracks the player successfully.

Also, we validate the proposed method in a real-life vehicle counting system. Fig. 9 (b) includes that our method correctly tracks a car, even the video stream acquired by a wireless consumer camera is at 6-10 fps.

However, the discriminative power of our method decreases when cameras move too quickly due to too much false motion alarms caused by quick motion of cameras. Fig. 9 (c) shows the movement of a student (lab2.avi is also taken by a wireless consumer camera just outside of our lab with 3-4 fps). In contrast with the two earlier cases, one great challenge involved in this video is that the camera is moving too quickly and our method tracks the player unsuccessfully. Moreover, another main limitation of our method is multi-object tracking. In fact, tracking multiple interactive objects itself would be a much more challenging problem. Multi-object tracking in frame-skipping videos is our future work.

**Figure 9.** (a) Results of tracking a player on football.avi (2-4 fps) with the motion of both the player and the camera. (b) Vehicle tracking in a real-life vehicle counting system (6-10 fps) using a wireless consumer camera. (c) Unsuccessful tracking when the camera moves too quickly (lab2.avi, 3-4 fps).
7. Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant No. 60903072; National Key Technology Research and Development Program of the Ministry of Science and Technology of China under Grant No. SQ2013SF13E00579.

Anlong Ming is with the Beijing Key Lab of Intelligent Telecom. Software and Multimedia, Beijing University of Posts and Telecomm., Beijing, 100876, China (e-mails: menganlong@bupt.edu.cn).

Huadong Ma is with Beijing Key Lab of Intelligent Telecom. Software and Multimedia, Beijing University of Posts and Telecomm., Beijing, 100876, China (e-mails: mhd@bupt.edu.cn).

Charles X. Ling is with the Department of Computer Science at The University of Western Ontario, London, Ontario N6A 5B7, Canada; (e-mails: cling@csd.uwo.ca).

8. References

[1] A. Ming, H. Ma, “Frame-skipping tracking for single object with global motion detection,” in Proceeding of the International Conference on Pattern Recognition (ICPR), pp. 1-4, Dec. 2008.
[2] W. Lao, J. Han and P. H.N. de With, “Automatic Video-Based Human Motion Analyzer for Consumer Surveillance System,” IEEE Trans. Consumer Electronics, vol. 55, no. 2, pp. 591-598, 2009
[3] A. Yilmaz, O. Javed, M. Shah, “Object Tracking: A Survey,” ACM Computer Survey, vol. 38, no. 4, pp. 45-50, 2006.
[4] I. F. Akyildiz, T. Melodia, K. R. Chowdhury, “A survey on wireless multimedia sensor networks,” Computer Networks, vol. 51, no. 4, pp. 921-960, 2007.
[5] L. Liu, X. Zhang, H. Ma, “Dynamic Node Collaboration for Mobile Target Tracking in Wireless Camera Sensor Networks,” IEEE INFOCOM 2009, pp. 1188-1196, 2009.
[6] C. Liu, H. Y. Shum and C. Zhang, “Hierarchical shape modeling for automatic face localization,” In: Proc. European Conf. Computer Vision, pp. 687-703, 2002.
[7] F. Porikli, O. Tuzel, “Object tracking in low-frame-rate video,” SPIE Image and Video Communications and Processing, vol. 5685, pp. 72-79, 2005.
[8] C. Benedek, T. Szirényi, “Bayesian Foreground and Shadow Detection in Uncertain Frame Rate Surveillance Videos,” IEEE Trans. Image Processing, vol. 17, no. 4, pp. 608-621, 2008.
[9] M. Han, A. Sethi, W. Hua and Y. Gong, “A detection-based multiple object tracking method,” In: Proc. IEEE International Conference of Image Processing, pp. 3065-3068, 2004.
[10] R. Kaucic, A. G. A. Perera, G. Brooksby, J. Kaufhold and A. Hoogs, “A unified framework for tracking through occlusions and across sensor gaps,” In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 990-997, 2005.

[11] G. Hua, Y. Wu, “Multi-scale visual tracking by sequential belief propagation,” In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 826-833, 2004.

[12] J. Sullivan, A. Blake, M. Isard and J. MacCormick, “Object localization by Bayesian correlation,” In: Proc. International Conf. on Computer Vision, pp. 1068-1075, 1999.

[13] K. Okuma, A. Taleghani, D. Freitas, J. J. Little and D. G. Lowe, “A boosted particle filter: Multi-target detection and tracking,” In Proc. European Conf. Computer Vision, pp. 28-39, 2004.

[14] Y. Li, H. Ai, T. Yamashita, S. Lao and M. Kawade, “Tracking in Low Frame Rate Video: A Cascade Particle Filter with Discriminative Observers of Different Life Spans,” IEEE Trans. PAMI, vol. 30, No. 10, pp. 1728-1740, 2008.

[15] M. Isard and A. Blake, “Condensation – conditional density propagation for visual tracking,” International Journal of Computer Vision, vol. 28, no. 1, pp. 5-28, 1998.

[16] J. W. Davis. “Hierarchical motion history images for recognizing human motion,” IEEE Workshop on Detection and Recognition of Events in Video, Vancouver, Canada, pp. 39-46, 2001.

[17] G. Bradski and J. Davis, “Motion Segmentation and Pose Recognition with Motion History Gradients,” International Journal of Machine Vision and Applications, Vol. 13, No. 3, pp. 174-184, 2002.

[18] D. Comaniciu, V. Ramesh and P. Meer, “Real-time tracking of non-rigid objects using mean shift,” In Proc. IEEE Conference on Computer Vision and Pattern Recognition, pp. 142-149, 2000.

[19] H. Zhou, Y. Yuan, C. Shi, “Object tracking using SIFT features and mean shift,” Computer Vision and Image Understanding, vol. 113, pp. 345-352, 2009.

[20] A. Lindquist, V. A. Yakubovich, “Optimal Damping of Forced Oscillations in Discrete-time Systems,” IEEE Trans. Automatic Control, vol. 42, no. 6, pp. 786-802, 1997.

[21] S. Arulampalam, S. Maskell, N. J. Gordon and T. Clapp, “A Tutorial on Particle Filters for On-line Nonlinear/Non-Gaussian Bayesian Tracking,” IEEE Trans. on Signal Processing, Vol. 50, no. 2, pp. 174-188, 2002.

[22] X. R. Li, V. P. Jilkov, “Survey of maneuvering target tracking. Part I: dynamic models,” IEEE Trans. Aerospace Electron. Syst. vol. 39, no. 4. 1333-1363, 2003.

Anlong Ming received a Ph.D. degree in Computer Science from the School of Computer at Beijing University of Posts and Telecommunications (BUPT), China, in 2008. In 2008 he joined Beijing Key Lab of Intelligent Telecommunications Software and Multimedia at Beijing University of Posts and Telecommunications, China. His current research focuses on multimedia systems and networking, grid computing and sensor networks.

Haudong Ma received his Ph.D. degree in Computer Science from the Institute of Computing Technology, Chinese Academy of Science in 1995. He is member of IEEE and ACM. He is a professor and director of Beijing Key Lab of Intelligent Telecommunications Software and Multimedia at Beijing University of Posts and Telecommunications, China. His current research focuses on multimedia systems and networking, grid computing and sensor networks.

Charles X. Ling earned his Ph.D. degree from the Department of Computer Science at the University of Pennsylvania in 1989. Since then he has been a faculty member in Computer Science at the University of Western Ontario. His main research areas include machine learning (theory, algorithms and applications), cognitive modelling and AI in general. He has published extensively in journals (such as Machine Learning, Journal of Artificial Intelligence Research and IEEE TKDE) and international conferences (such as IJCAI and ICML). He is also the director of Data Mining Lab, leading data mining development in CRM, bioinformatics and the Internet. He has managed several data-mining projects for major banks and insurance companies in Canada.