REAL-TIME OBJECT TRACKING BASED ON THE RELATIVE HIST MODEL WITHIN PARTICLE FILTER FRAMEWORK

LingFeng Wang, ChunHong Pan

National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences
{lfWang@nlpr.ia.ac.cn, chpan@nlpr.ia.ac.cn}

ABSTRACT

This paper proposes a novel real-time tracking algorithm by using the relative hist model within particle filter framework. The target-region is roughly enclosed with a rectangle as usual, and color features are used to describe all the types of regions by calculating their color histograms. Inevitably, the background pixels are included in the target-region and the histogram of target-region may be corrupted when performing tracking algorithm. Even the target fails to track. Thus, the relative hist model is proposed to reduce the influence of background pixels. In this model, we not only consider the similarity between the candidate-region and target-region, but also consider the similarity between the candidate-region and background. In other words, the relative hist model tries to find a candidate-region which is more similar to the target-region but less similar to the background. By adopting this model, our tracking algorithm can accurately track the object in real-time. Experiments are performed in various tracking scenes. The experiment results show that our algorithm is of appealing with respect to robustness for real-time object tracking against various backgrounds.

***Index Terms*—** Relative hist model, Particle Filter

1. INTRODUCTION

Real-time tracking is a fundamental task for various applications such as surveillance, vision-based control, human-computer interfaces, and so on. In general, real-time tracking methods can be mainly classified into two categories: either based on the detecting and tracking of a sparse collection of features (feature-based trackers, such as Kanade-Lucas-Tomasi tracker [1]), or based on minimizing the sum of squared differences between two corresponding regions (blob-based trackers, i.e. mean-shift tracker[2, 3], kalman filter tracker [4, 5], and particle filter tracker [5, 6, 7]).

The strategy of real-time tracking, either the feature-based tracking or the blob-based tracking, is to search a candidate-region in the search-region between consecutive frames (Fig.1). For the blob-tracking, a regular shape, i.e. a rectangular window, and color histogram from the enclosed region are perhaps the simplest but effective way to represent the appearance of target-region. However, this inevitably includes background pixels when the foreground shape cannot be closely approximated. As shown in the Fig.1, the target-region is composed by the foreground-target and in-target-region's background. Accordingly, color histogram of target-region can be corrupted by the background pixels and the corresponding tracking result will be unstable, or even failed. Thus, it is important to approximate the foreground-target exactly or reduce the influence of background pixels significantly. Many methods are proposed to solve the problem. In many applications, the foreground-target is always gained by background subtraction methods, i.e. the mixture of Gaussians(**MoG**) proposed in [8]. However, these methods do not work well in the case of un-stationary camera. In order to approximate the foreground-target under the un-stationary camera, segmentation algorithms, such as graph cut in [9], are frequently applied when performing tracking. Although these methods perform well in many scenes, the main limitation is that both memories and computation costs increase seriously with the increasing of the images resolution. That is to say, real-time is a common problem by these algorithm in many applications.

The core of particle filter algorithm contains two models, i.e. the dynamic model and observation model. Traditionally, the observation likelihood (observation model) is represented by the similarity between the target-region and candidate-region. The problem arising from these algorithms is that, the similarity between the candidate-region and background is not take into consideration. Thus, when the target-region contains background pixels, the observation likelihood may be not calculated exactly. Even the target fails to track.

In this paper, a novel relative hist model within the particle filter framework is presented in order to reduce the influence of background pixels. The observation likelihood is represented by the novel relative hist model. In this model, we consider the observation likelihood as the probability that a candidate-region belongs to target-region but not the simple similarity. Each candidate-region has two relative probabilities, i.e. belongs to target-region or belongs to background. The purpose of relative hist model is to gain the candidate-region which has max probability of belonging to target-region but not belonging to background. Since all the probabilities are represented by the similarity, the relative hist model tries to find a candidate-region that is more similar to the target-region but less similar to the background. Accordingly, the similarity between the candidate-region and target-region as well as the similarity between the candidate-region and background are taken into consideration. When implementing our algorithm, the target-region is roughly enclosed with a rectangle, and color features are used to describe all the types of regions by calculating theirs color histogram, and the Bhattacharyya coefficient [10] is used to measure the similarity between regions. Finally, experiments with various scenes are performed to test our algorithm, and results show that our algorithm is of appealing with respect to robustness.

This paper is organized as follows: our novel blob-tracking algorithm are described in Section 2 in detail; experiment results on real data compared with the traditional methods are reported in Section 3; the conclusive remark is addressed at the end of this paper.

This research is sponsored by the National '863' High-Tech Program of China under the grant No. 2009AA012104 and Natural Science Foundation of China under the grant Nos: 60675012, 60873161.

978-1-4244-4296-6/10/$25.00 ©2010 IEEE

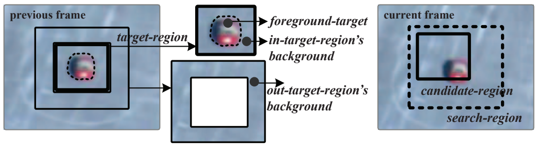

Fig. 1. The figure describes all types of regions. The target-region is composed by foreground-target and in-target-region background. The background includes in-target-region's background and out-target-region's background.

2. METHOD DESCRIPTION

We present the details of the proposed tracking algorithm in this section. First, the particle filter framework is presented in Subsection 2.1. Then, the details of the observation and state of our algorithm is presented in Subsection 2.2. After that, the relative hist model as observation model and gaussian model as dynamic model are illustrated in Subsection 2.3 and 2.4, respectively.

2.1. Particle Filter Algorithm

The Bayesian estimate is used to recursively estimate a time evolving posterior distribution. This distribution describes the object state by given all observations. Denote $\mathbf{z}_t, \mathbf{x}_t$ as the evolution of observation and state at time t, respectively. Two models are defined as follow. The first one is the dynamic model, which calculates the evolution of the state $\{\mathbf{x}_t, t \in \mathbf{T}\}$, given by

$$\mathbf{x}_t = \mathcal{F}_t(\mathbf{x}_{t-1}, \mathbf{w}_{t-1}) \tag{1}$$

and the second one is the observation model, which recursively estimates $\mathbf{x}_t$, given by

$$\mathbf{z}_t = \mathcal{H}_t(\mathbf{x}_t, \mathbf{v}_t) \tag{2}$$

where $\mathcal{F}_t$ and $\mathcal{H}_t$ are nonlinear functions, $\mathbf{w}_t$ and $\mathbf{v}_{t-1}$ are independent and identically distributed noise process.

From the Bayesian perspective, the tracking problem is actually to estimate the filter distribution recursively, i.e. to calculate the belief of the state $\mathbf{x}_t$ at time t by given observation $\mathbf{z}_{1:t}$. Thus, we need to calculate the probability density function $\mathrm{p}(\mathbf{x}_t|\mathbf{z}_{1:t})$. We assume $\mathrm{p}(\mathbf{x}_0|\mathbf{z}_0) = \mathrm{p}(\mathbf{x}_0)$ as initial probability density function. Then, $\mathrm{p}(\mathbf{x}_t|\mathbf{z}_{1:t})$ may be obtain recursively by two stages: **prediction** and **update**.

PREDICTION: Suppose the required $\mathrm{p}(\mathbf{x}_{t-1}|\mathbf{z}_{1:t-1})$ at time $t-1$ is available, the prediction stage obtained via the Chapman-Kolmogorov formula:

$$\mathrm{p}(\mathbf{x}_t|\mathbf{z}_{1:t-1}) = \int \mathrm{p}(\mathbf{x}_t|\mathbf{x}_{t-1})\mathrm{p}(\mathbf{x}_{t-1}|\mathbf{z}_{1:t-1})\mathrm{d}\mathbf{x}_{t-1} \tag{3}$$

where the probabilistic $\mathrm{p}(\mathbf{x}_t|\mathbf{x}_{t-1})$ is defined by Equation (1).

UPDATE: When the measurement $\mathbf{z}_t$ at time t is available, we update the prior through the Bayesian rule,

$$\mathrm{p}(\mathbf{x}_t|\mathbf{z}_{1:t}) \propto \mathrm{p}(\mathbf{z}_t|\mathbf{x}_t)\mathrm{p}(\mathbf{x}_t|\mathbf{z}_{1:t-1}) \tag{4}$$

where the likelihood $\mathrm{p}(\mathbf{z}_t|\mathbf{x}_t)$ is defined by Equation (2).

Particle filter[5, 6, 7] algorithm obtains the Bayesian estimate by Monte Carlo simulations, and describes $\mathrm{p}(\mathbf{x}_t|\mathbf{z}_{1:t})$ by a set of N particles, and associates them with weights $\{\mathbf{x}_t^k, \boldsymbol{\omega}_t^k\}_{k=1}^{\mathrm{N}}$. Then, the posterior density at time t can be approximated as follow,

$$\mathrm{p}(\mathbf{x}_t|\mathbf{z}_{1:t}) = \sum_{k=1}^{\mathrm{N}} \boldsymbol{\omega}_t^k \delta\Big(\mathbf{x}_t - \mathbf{x}_t^k\Big) \tag{5}$$

where $\delta(.)$ is the dirac function and the weights are normalized by the equation $\sum_k \boldsymbol{\omega}_t^k = 1$.

2.2. Details on Observation $\mathbf{z}_t$ and State $\mathbf{x}_t$

In our experiment, the target-region is roughly enclosed with a rectangle, and color features are used to describe all the types of regions by calculating theirs color histogram $\mathbf{q} = \{q_i\}$, $\sum_i q_i = 1$. The histogram of target-region, candidate-region and background are denoted as $\mathbf{q}^o$, $\mathbf{q}^c$, and $\mathbf{q}^b$, respectively. Meanwhile, the Bhattacharyya coefficient [10] is used to measure the similarity between two regions $\mathbf{q}^1$ and $\mathbf{q}^2$, given by,

$$\rho(\mathbf{q}^1, \mathbf{q}^2) = \sum_i \sqrt{q_i^1 q_i^2} \tag{6}$$

Accordingly, observation $\mathbf{z}_t$ is represented by the histogram $\mathbf{q}$. Since we only record the position and size of a region, the state $\mathbf{x}_t$ is represented by $\{\mathbf{p} = (p_x, p_y), \mathbf{s} = (h, w)\}$, where $\mathbf{p}$, $\mathbf{s}$ are the center position and size of the region respectively.

2.3. Relative Hist Model as Observation Model

As described in subsection 2.1, the k-th particle weight is updated by specifying prior density function as the importance density,

$$\boldsymbol{\omega}_t^k = \boldsymbol{\omega}_{t-1}^k \mathrm{p}(\mathbf{z}_t|\mathbf{x}_t^k) \tag{7}$$

where $\boldsymbol{\omega}_{t-1}^k$ is prior importance weight at time $t-1$ and $\mathrm{p}(\mathbf{z}_t|\mathbf{x}_t^k)$ is the observation likelihood at state $\mathbf{x}_t^k$.

Traditionally, the observation model (likelihood) is represented by the similarity between the target-region and candidate-region. Since all types of regions are represented by the histograms, this similarity is defined by the Bhattacharyya coefficient, give by

$$\boldsymbol{\omega}_t^k = \boldsymbol{\omega}_{t-1}^k \mathrm{p}(\mathbf{z}_t|\mathbf{x}_t^k) = \boldsymbol{\omega}_{t-1}^k \rho(\mathbf{q}_{t-1}^o, \mathbf{q}_t^{c_k}) \tag{8}$$

where $\mathbf{q}_{t-1}^o$, $\mathbf{q}_t^{c_k}$ are the histogram of target-region at time $t-1$ and the histogram of k-th candidate-region at time t.

The update Equation(8) works well when the target-region is only composed by the foreground-target. However, its inevitably includes background pixels when the target-region is enclosed with a rectangle roughly. Therefore, this similarity may not exactly describe the probability that the k-th candidate-region is target-region. For an instance, when the background ratio of the target-region more than half, the similarity describes more about the similarity between the candidate-region and background. The main reason is that the traditional method only consider the similarity between candidate-region and target-region, but not take the similarity between candidate-region and background into consideration. Thus, different from the traditional method, we define the observation likelihood as the probability that a candidate-region belongs to target-region $p(o|\mathbf{q}_t^{c_k})$. Accordingly, the weights $\boldsymbol{\omega}_t^k$ updated by

$$\begin{aligned}
\boldsymbol{\omega}_t^k &= \boldsymbol{\omega}_{t-1}^k \mathrm{p}(\mathbf{z}_t|\mathbf{x}_t^k) \\
&= \boldsymbol{\omega}_{t-1}^k \mathrm{p}(o|\mathbf{q}_t^{c_k}) \qquad \textbf{\textit{Bayesian}} \\
&= \boldsymbol{\omega}_{t-1}^k \frac{\mathrm{p}(\mathbf{q}_t^{c_k}|o)\mathrm{p}(o)}{\mathrm{p}(\mathbf{q}_t^{c_k}|o)\mathrm{p}(o) + \mathrm{p}(\mathbf{q}_t^{c_k}|b)\mathrm{p}(b)} \\
&= \boldsymbol{\omega}_{t-1}^k \frac{\mathrm{p}(\mathbf{q}_t^{c_k}|o)}{\mathrm{p}(\mathbf{q}_t^{c_k}|o) + \mathrm{p}(\mathbf{q}_t^{c_k}|b)\lambda} \qquad \lambda = \frac{\mathrm{p}(b)}{\mathrm{p}(o)}
\end{aligned} \tag{9}$$

where, the $\mathrm{p}(\mathbf{q}_t^{c_k}|o)$ denotes that the probability that the histogram of target-region in current frame is $\mathbf{q}_t^{c_k}$ when the candidate-region belongs to target-region. Same as the definition of $\mathrm{p}(\mathbf{q}_t^{c_k}|o)$, the $\mathrm{p}(\mathbf{q}_t^{c_k}|b)$ denotes that the probability that the histogram of background is $\mathbf{q}_t^{c_k}$ when the candidate-region belongs to background.

From the definition of $\mathrm{p}(\mathbf{q}_t^{c_k}|o)$ and $\mathrm{p}(\mathbf{q}_t^{c_k}|b)$, the $\mathrm{p}(\mathbf{q}_t^{c_k}|o)$ can be calculated by the similarity between the candidate-region and target-region, while the $\mathrm{p}(\mathbf{q}_t^{c_k}|b)$ can be calculated by the similarity between the candidate-region and background. That is, $\mathrm{p}(\mathbf{q}_t^{c_k}|o) = \rho(\mathbf{q}_{t-1}^o, \mathbf{q}_t^{c_k})$ and $\mathrm{p}(\mathbf{q}_t^{c_k}|b) = \rho(\mathbf{q}_{t-1}^b, \mathbf{q}_t^{c_k})$. From Equation(9), the coefficient $\lambda = \frac{\mathrm{p}(b)}{\mathrm{p}(o)}$ defines the prior density that a candidate-region belongs to the background relative to target-region. Based on the continuity of the movement, we assume that the position and size recursive frames vary small. Thus, $\mathrm{p}(o)$ is defined by the weight of previous particle, that is $\mathrm{p}(o) = \boldsymbol{\omega}_{t-1}^k$. Since we have no prior density of background, the probabilities that the candidate-region belong to the background is defined as the uniform distribution, that is $\mathrm{p}(b) = \frac{1}{\mathrm{N}}$. Accordingly the coefficient λ is given by

$$\lambda = \frac{\mathrm{p}(b)}{\mathrm{p}(o)} = \frac{1}{\mathrm{N}\boldsymbol{\omega}_{t-1}^k} \tag{10}$$

where N is the number of particles. At last, from the Equation(9,10) and the definition of $\mathrm{p}(\mathbf{q}_t^{c_k}|o)$, $\mathrm{p}(\mathbf{q}_t^{c_k}|b)$, we can gain that the k-th particle weight $\boldsymbol{\omega}_t^k$ is updated as follow,

$$\boldsymbol{\omega}_t^k = \boldsymbol{\omega}_{t-1}^k \frac{\rho(\mathbf{q}_{t-1}^o, \mathbf{q}_t^{c_k})}{\rho(\mathbf{q}_{t-1}^o, \mathbf{q}_t^{c_k}) + \frac{\rho(\mathbf{q}_{t-1}^o, \mathbf{q}_t^b)}{\mathrm{N}\boldsymbol{\omega}_{t-1}^k}} \tag{11}$$

From the Equation(11), the k-th particle weight is updated by the similarity between the candidate-region and target-region relative to the similarity between the candidate-region and background. Thus, we denote this observation model as relative hist model.

2.4. Gaussian Model as Dynamic Model

As described in subsection 2.2, the state is represented by the position and size. So, the state $\mathbf{x}_t^k$ of each particle is represented with the position $\mathbf{p}_t^k$ and size $\mathbf{s}_t^k$, that is, $\mathbf{x}_t^k = [\mathbf{p}_t^k \; \mathbf{s}_t^k]^{\mathrm{T}}$. Generally, gaussian model is adopted as dynamic model, given by

$$\begin{aligned} \mathbf{p}(\mathbf{x}_t^k|\mathbf{x}_{t-1}^k) &= \mathcal{N}(\mathbf{x}_{t-1}^k + \frac{\partial \mathbf{x}_{t-1}^k}{\partial t}, \boldsymbol{\sigma}_{\mathbf{x}}) \\ &= \mathcal{N}(\begin{bmatrix}\mathbf{p}_{t-1}^k \\ \mathbf{s}_{t-1}^k\end{bmatrix} + \frac{\partial \begin{bmatrix}\mathbf{p}_{t-1}^k \\ \mathbf{s}_{t-1}^k\end{bmatrix}}{\partial t}, \begin{bmatrix}\boldsymbol{\sigma}_{\mathbf{p}} \\ \boldsymbol{\sigma}_{\mathbf{s}}\end{bmatrix}) \end{aligned} \tag{12}$$

where $\boldsymbol{\sigma}_{\mathbf{p}}$, $\boldsymbol{\sigma}_{\mathbf{s}}$ are the variance of position and size, and $\mathcal{N}(.)$ is the normal distribution function.

3. EXPERIMENTS AND RESULTS

In the following experiments, a multi-color is adopted as observation model based on Hue-Saturation-Value color histograms[11], the number of particles is selected as $\mathrm{N} = 50$, the variance of position and size are selected as $\boldsymbol{\sigma}_{\mathbf{p}} = [10 \; 10]^{\mathrm{T}}$, $\boldsymbol{\sigma}_{\mathbf{s}} = [4 \; 4]^{\mathrm{T}}$, and the initial position and size are initialed at first frame.

The overview of our algorithm is illustrated in Fig.2. A previous frame is processed at first by calculating two histograms $\mathbf{q}^o$ and $\mathbf{q}^b$. Then, for each candidate-region, the observation likelihood is initialed as the uniform distribution, that is $\frac{1}{\mathrm{N}}$, and updated by Equation(11) after the calculation of histogram $\mathbf{q}^{c_k}$. Meanwhile, the prediction model is calculated by the Equation(12). Then, the posterior distribution is estimated by Equation(5). Finally, we can get the position and size of the target-region. Meanwhile, the histogram of the target-region is updated by

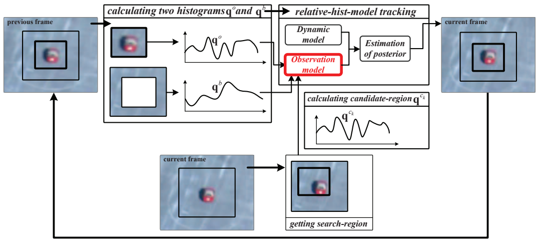

Fig. 2. Overview of our novel algorithm. The core of the algorithm is the observation model boxed with bold red frame.

$$\mathbf{q}_t^o = (1-\eta) \cdot \mathbf{q}_{t-1}^o + \eta \cdot \mathbf{q}_t^o \tag{13}$$

where η is the updating ratio (in our experiments $\eta = 0.05$).

Both visual and numerical methods are used to evaluate our algorithm by comparing with the traditional methods and the ground truth. First, the comparison between our algorithm and traditional methods is illustrated in Fig.3. Then, the results of our algorithm compared with the ground truth are illustrated in the Fig.4. Fig.4(b,c) show the variations of the error ϵ, **False Negatives** and **False Positives** along with the variations of the size of the initial target-region. Error ϵ is defined by the difference of two center points, given by $\epsilon = \frac{\sum_t \|c_t - \hat{c}_t\|^2}{\mathrm{T}}$, where c_t and $\hat{c}_t$ are the center point of the output target-region of our algorithm and the ground truth, and T is total frame. **False Negatives** and **False Positives** are defined as $\frac{\sum_t \frac{\hat{r}_t - r_t \cap \hat{r}_t}{\hat{r}_t}}{\mathrm{T}}$, $\frac{\sum_t \frac{r_t - r_t \cap \hat{r}_t}{\hat{r}_t}}{\mathrm{T}}$ respectively, where r_t and $\hat{r}_t$ are the output target-region of our algorithm and the ground truth, respectively, and the operation $\cap$ is denoted to gain the intersection region. As illustrated in Fig.3, when the initial target-region is small, the result of our algorithm is same as the traditional one. However, when the initial target-region is large, that is the target-region contains a lot background pixels, our algorithm can still track the target but the traditional one can not. Furthermore, the target-region gradually closes to the foreground-target by our algorithm. As shown in Fig.4(b,c), when the initial target-region is increasing, error ϵ and **False Positives** are quite small and increasing slowly, and **False Negatives** is stable. Accordingly, we can get the conclusions that our algorithm is less sensitive to the quality of initialization than the traditional one. The main reason is that the influence of background pixels on the target-region representation is eliminated by the relative hist model.

In the Fig.5, the performance of our algorithm is tested on other scenes. Although the acquisition equipment are difference (i.e. fisheye in the first scene and aerial cameras in the third scene), and the illumination are changing (i.e. indoor in the first and second scenes and outdoor in the third scene), all the target-region can still robust to track by our algorithm.

The speed time of our algorithm is also measured. We use a standard PC with a **1.8** GHz processor and **512** MB of memory, and choose the **C++** in our experiments. As illustrated in Fig.6(a), the processing time is linearly proportional to the initial target-region's size, and in Fig.6(b) the max processing time is less than **0.1**s. This makes our algorithm suites to real-time processing systems.

4. CONCLUSIONS

In this paper, we present a novel relative hist model in real-time tracking. The main contributions of this model is that it consider

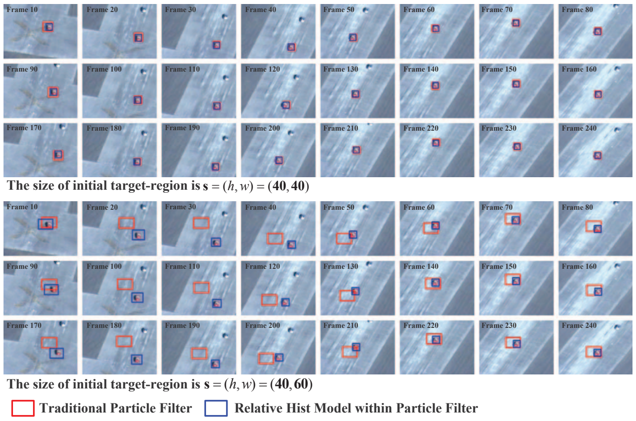

Fig. 3. Tracking results in the same scene with different initial target-region size compared with the traditional algorithm.

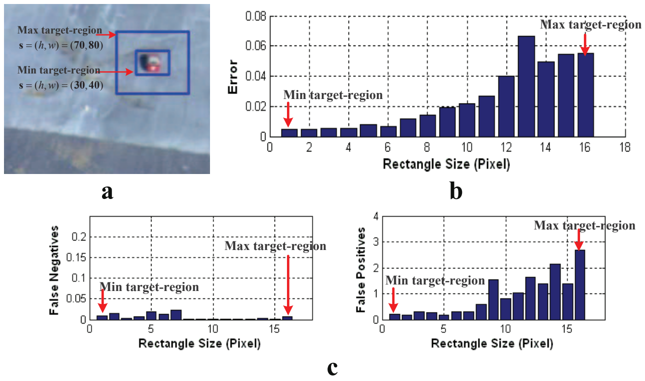

Fig. 4. Comparing our results with ground truth. Fig(a) shows the initial size of target-region. Only the max and min size of target-region are drawn in figure.

the observation likelihood as the probability that a candidate-region belongs to the target-region but not the simple similarity. Accordingly, the influence of background pixels can be reduced. At last, the relative hist model is embedded with the particle filter framework when performing tracking. Experimental results show that our method is robust and suitable for many scenes.

5. REFERENCES

[1] Carlo Tomasi and Takeo Kanade, "Shape and motion from image streams: a factorization method - part 3 detection and tracking of point features," Tech. Rep. CMU-CS-91-132, CMU, School of Computer Science, 1991.

[2] Robert T. Collins, "Mean-shift blob tracking through scale space," in *CVPR*, 2003, pp. 234–240.

[3] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 564–577, 2003.

[4] A.G.C. Plant, J.B. Chan, and Y.T. Hu, "A kalman filter based tracking scheme with input estimation," *Aerospace and Electronic Systems*, pp. 237–244, 1979.

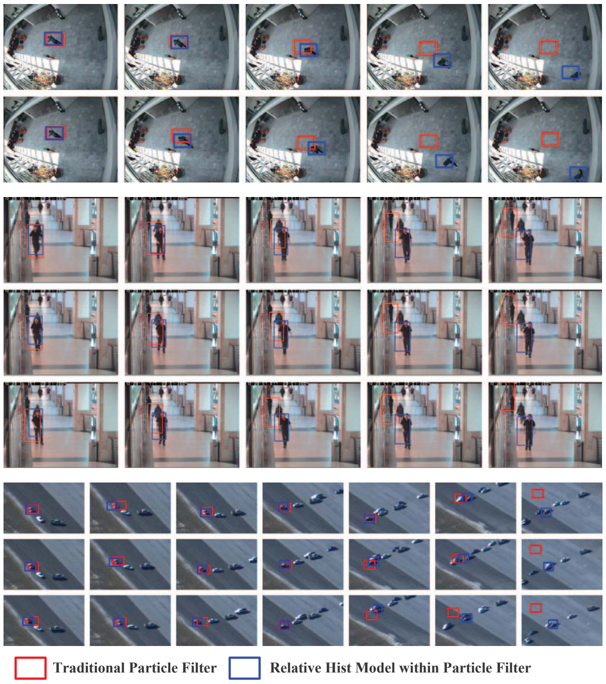
Traditional Particle Filter
Relative Hist Model within Particle Filter

Fig. 5. More tracking results compared with traditional algorithm.

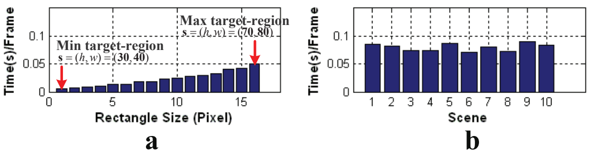

Fig. 6. Fig(a) shows the time variations along with the variations of the initial target-region's size in the same scene. Fig(b) shows the average processing time with different **10** scenes.

[5] S. Arulampalam B. Ristic and N. Gordon, "Beyond the kalman filter: Particle filters for tracking applications," 2004.

[6] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussianbayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, pp. 174–188, Feb. 2002.

[7] Jaco Vermaak, Arnaud Doucet, and Patrick Pérez, "Maintaining multi-modality through mixture tracking," in *ICCV*. 2003, pp. 1110–1116, IEEE.

[8] W. Grimson C. Stauffer, "Adaptive background mixture models for real-time tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1999, vol. 2, pp. 246–252.

[9] S M Shahed Nejhum, Jeffrey Ho, and Ming-Hsuan Yang, "Visual tracking with histograms and articulating blocks," in *CVPR*, 2008, pp. 1–8.

[10] Jianhua Lin, "Divergence measures based on the shannon entropy," *IEEE Transactions on Information Theory*, vol. 37, pp. 145–151, 1991.

[11] P. Prez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," in *In Proc. ECCV*, 2002, pp. 661–675.