Multiple objects tracking method based on particle filter

BUDI SUGANDI, HYOUNGSEP KIM, JOO KOOI TAN and SEIJI ISHIKAWA
Department of Control Engineering
Kyushu Institute of Technology
1-1, Sensui, Tobata, Kitakyushu 804-8550
JAPAN
kim@cntl.kyutech.ac.jp

Abstract: - Object tracking is a challenging problem due to the presence of noise, occlusion, clutter and dynamic change in the scene other than the motion of the object of interest. A variety of tracking algorithms has been proposed and implemented to overcome the related difficulties, but there are still some problems need to be covered. In this paper, we present an approach for multiple objects tracking based on particle filter algorithm. We use the particle filter to predict the trajectory of the target. The problem of occlusion is predicted based on the likelihood measurement and estimated samples distance. The particle filter approximates a posterior probability density of the state using samples or particles. Each state is denoted as the hypothetical state of the tracked object and its weight which is predicted based on the system model. In this paper, the state is treated as a position, speed, size, scale and appearance of the object. The samples weight is considered as the likelihood of each particle which is measured based on the similarity between the colour feature of the target model and the objects. And finally, the mean state of the particles is treated as the estimated state of the object. The experiments are performed to confirm the effectiveness of the method to track multiple objects.

Key-Words: - multiple objects tracking; object occlusion; particle filter; color feature

1 Introduction

The increasing interest in the object tracking is motivated by a huge number of promising applications that can now be tackled in real-time applications. These applications include performance analysis, surveillance, video-indexing, smart interfaces, teleconferencing and video compression, etc. However, tracking object can be extremely complex and time consuming especially when it is performed in outdoor environments. The problems of object tracking in outdoor environments include fake-motion background, illumination changes, shadows and presence of clutter. Some problems are also taken into account concerning whether a single object or multiple objects should be tracked. Because the multiple objects tracking has more challenging problem due to the presence of objects occlusion, objects appearing and disappearing. A variety of tracking algorithms have been proposed and implemented to overcome these difficulties.

Recently, many researches are performed the tracking algorithm based on stochastic methods using the state space to model the underlying dynamics of the tracking system such as Kalman filter [1-2] and particle filter [3-6]. Kalman filter is a common approach for dealing with target tracking in the probabilistic framework. In a linear-Gaussian model with linear measurement, there is always only one mode in the posterior probability density function (pdf). The Kalman filter can be used to propagate and update the mean and covariance of the distribution of the model [2]. However, it cannot resolve the tracking problem when the model is nonlinear and non-Gaussian [7]. The extended Kalman filter can deal to this problem, but still has a problem when the nonlinearity and non-Gaussian system cannot be approximated accurately. The particle filter has been introduced and become popular algorithm to handle the estimation problem of nonlinear and non-Gaussian framework. The particle filter, also known as sequential Monte Carlo [4], is the most popular approach which recursively constructs the posterior pdf of the state space using Monte Carlo integration. It approximates a posterior probability density of the state such as the object position by using samples or particles. The probability distribution of the state of the tracked object is approximated by a set of particles, which each state is denoted as the hypothetical state of the tracked object and its weight. It has been developed in the computer vision community and applied to tracking problem and is also known as the

ISBN: 978-1-61804-062-6

Condensation algorithm [3]. For another particle filter, Bayesian bootstrap filter was introduced [5].

Although particle filters have been widely used in recent years, they have important drawbacks such as samples are spread around several modes pointing out the different hypotheses in the state space [8]. Especially in the multiple objects tracking, the objects with higher likelihood may monopolize the samples set and objects whose samples exhibit lower likelihood have higher probability of being lost. Besides that, the multiple objects tracking has another problem with occlusion, object appearing and disappearing and appearance change.

Many improvements have been introduced to overcome those problems, but there are still much issues need to be covered. Different approaches have been taken in order to overcome these facts. Nummiaro et al. [9] used a particle filter based on color histograms features. Histograms are robust to partial occlusions and rotations, however, no multiple targets tracking are considered and complete occlusions were not handled. Perez et al. [10] proposed also a particle filter based on color histogram. They introduced interesting extension in multiple-part modeling, incorporation of background information and multiple targets tracking. Nevertheless, it required an extremely large number of samples, since one sample contains information about the state of all targets, dramatically increasing the state dimensionality. Furthermore, no appearance model updating was performed, what usually leads to target loss in dynamic scenes. Comaniciu et al. [11] approach relied on gradient-based optimization and color-based histograms. In this case, no dynamic model was used in their approach therefore no occlusion can be predicted. Deutscher et al. [12] presented an interesting approach called annealing particle filter which aims to reduce the required number of samples, however, it could be inappropriate in a cluttered environment. They combine edge and intensity measures but they focused on motion analysis, and thus, no occlusion handling is explored.

Another issue is on the multiple objects tracking problem that is the management of multiple tracks caused by newly appearing objects and the disappearing of already existing objects. Some of them rely on hybrid sequential state estimation. In [13], the state vector denoting all the existing targets is augmented by a discrete random variable which represents the number of existing objects in a video sequence. The particle filter developed in [14] has multiple models for the object motion, and comprises an additional discrete state component, denoting which of the motion models is active. The Bayesian Multiple-Blob Tracker (BraMBLe) [15] presented a multiple persons tracking system based on statistical appearance models. The multiple blob tracking is managed by incorporating the number of objects present in the state vector and state vector is augmented as in [13] when a new object enters the scene.

In this paper, we introduce a method based on particle filter to solve some tracking problems related to the difficulties described above, such as multiple objects tracking in the presence of occlusion.

The remaining of the paper is organized as following. In section 2, we describe our approach to track the moving object using our proposed method. Some experimental results are presented in section 3 and finally conclusions and future works are presented in section 4.

2 Tracking Method

In this paper, we proposed an algorithm based on particle filtering in conjunction with a color feature to track multiple objects appear in the scene. We considered the motion of the central point of a bounding box as the target model using first-order dynamics model. The state of the object is defined as $\mathbf{s}_k = (\mathbf{x}_k, \dot{\mathbf{x}}_k, \mathbf{w}_k, \dot{\mathbf{w}}_k, \mathbf{A}_k)$ where the components are position, speed, bounding-box size, bounding-box scale and pixel appearance, respectively. The observations $\boldsymbol{z}_k$ is given by input images $\boldsymbol{I}_k$. Firstly, the initialization of the samples is done in the first frame. Next, the samples are predicted based on a system/transitional model by propagating each sample based on the transitional model. The samples update is performed based on the observation model. In this paper, we use color distribution of the object as the observation model. Then using the Bhattacharya distance, the similarity between the color distribution of the target and the samples can be measured. Based on the Bhattacharya distance, the weight of each sample is measured. The target state estimation is performed based on the sample's weight. The resampling is performed for the next sample iteration to generate a new samples set. During the resample step, samples with a high weight may be chosen several times leading to identical copies, while others with relatively low weights may be ignored and deleted. And finally, the target model update is performed adaptively based on the best match of the target model. The object occlusion is predicted based on likelihood measurement and number appearance of the samples. The detail of each process is described as following.

2.1 Tracking System Model

We consider the motion of the object as the discrete time 2-dimensional (2D) motion model. The state vector at a time step k is denoted by $\boldsymbol{s}_k$, including position, speed, bounding box size and bounding box scale of each sample and is predicted according to the following expressions,

$$\begin{aligned}\hat{\mathbf{x}}_k &= \mathbf{x}_{k-1} + \dot{\mathbf{x}}_{k-1}\Delta t + \xi_{\mathbf{x}},\\ \hat{\dot{\mathbf{x}}}_k &= \dot{\mathbf{x}}_{k-1} + \xi_{\dot{\mathbf{x}}},\\ \hat{\mathbf{w}}_k &= \mathbf{w}_{k-1} + \dot{\mathbf{w}}_{k-i}\Delta t + \xi_{\mathbf{w}},\\ \hat{\dot{\mathbf{w}}}_k &= \dot{\mathbf{w}}_{k-1} + \xi_{\dot{\mathbf{w}}}.\end{aligned} \quad (1)$$

Here, $\hat{x}_k, \hat{\dot{x}}_k, \dot{w}_k, \hat{\dot{w}}_k$ are the position, speed, bounding box and bounding box scale of each sample state, respectively. The random vectors $\xi_x, \xi_{\dot{x}}, \xi_w, \xi_{\dot{w}}$ are Gaussian noise providing the system with a diversity of hypotheses. Each component of the samples is predicted by propagating the samples according to this transition model.

2.2 Observation Model

We used a color distribution of the object as observation model which is obtained by building the color histogram in the RGB color space. The color distribution at location $\mathbf{y}$ is calculated as follows

$$p_{\mathbf{y}}^{(u)} = f\sum_{j=1}^{I} g\left(\left\|\mathbf{y}-\mathbf{x}_j\right\|/a\right)\delta\left[h\left(\mathbf{x}_j\right)-u\right] \quad (2)$$

where, u is number of bins, I is the number of pixels in the region, $\boldsymbol{x}_j$ is the position of pixels in the region, δ is the Kronecker delta function, a is the normalization factor, f is the scaling factor to ensures that $\sum_{u=1}^{m} p_{\mathbf{y}}^{(u)} = 1$, $g(.)$ is weighting function and $h(.)$ is assigned color bin at some position, respectively.

The similarity between two color histograms can be calculated using Bhattacharya distance $d = \sqrt{1-\rho[p,q]}$, where $\rho[p,q] = \sum_{u=1}^{m}\sqrt{p^{(u)}q^{(u)}}$. Similar histogram will have a small Bhattacharya distance which corresponds to high sample weight. Then, the weight $\pi^{(i)}$ of sample state $\boldsymbol{x}^{(i)}$ is calculated according to,

$$\pi^{(i)} = \frac{1}{\sqrt{2\pi\sigma}}\exp(-(1-\rho[p(\mathbf{x}^{(i)}),q])/2\sigma^2). \quad (3)$$

Where $p(x^{(i)})$ and q are the color histogram of the samples and target, respectively.

2.3 State Estimation

The target state estimation is computed according to the following expressions

$$\begin{aligned}\mathbf{x}_k &= (1-\alpha_{\mathbf{x}})(\mathbf{x}_{k-1} + \dot{\mathbf{x}}_{k-1}\Delta t) + \alpha_{\mathbf{x}}\sum_{i=1}^{N}\bar{\pi}_k^i\hat{\mathbf{x}}_k^i,\\ \dot{\mathbf{x}}_k &= (1-\alpha_{\dot{\mathbf{x}}})\dot{\mathbf{x}}_{k-1} + \alpha_{\dot{\mathbf{x}}}(\mathbf{x}_k - \mathbf{x}_{k-1})/\Delta t,\\ \mathbf{w}_k &= (1-\alpha_{\mathbf{w}})\mathbf{w}_{k-1} + \alpha_{\mathbf{w}}\sum_{i=1}^{N}\bar{\pi}_k^i\hat{\mathbf{w}}_k^i,\\ \dot{\mathbf{w}}_k &= (1-\alpha_{\dot{\mathbf{w}}})\dot{\mathbf{w}}_{k-1} + \alpha_{\dot{\mathbf{w}}}(\mathbf{w}_k - w_{k-1})/\Delta t.\end{aligned} \quad (4)$$

Here, $\alpha_{\mathbf{x}}, \alpha_{\dot{\mathbf{x}}}, \alpha_{\mathbf{w}}, \alpha_{\dot{\mathbf{w}}} \in [0,1]$ denote the adaptation rates.

2.4 Resample

The resample step is performed for the next sample iteration to generate a new samples set. During the resample step, samples with a high weight may be chosen several times leading to identical copies, while others with relatively low weights may be ignored and deleted. The resample step can be done in several different ways. One straightforward way is as the following steps [16]:

1. Generate a random number r that is uniformly distributed on [0, 1].
2. Calculate the normalized cumulative probability

$$c_k^{(0)} = 0,\ c_k^{(i)} = c_k^{(i-1)} + \pi_k^{(i)},\ c_k'^{(i)} = \frac{c_k^{(i)}}{c_k^{(N)}}$$

3. by binary search, find the smallest j for which $c_k' \geq r$ and set the new particle $x_k^i = x_k^j$

2.5 Subsection

The target appearance must be updated to adapt the object appearance change. In this paper, the target model is only updated when the target is not occluded by another object and the likelihood of the estimated target's state suggests that the estimate is sufficiently reliable. In this case, it is updated using an adaptive filter

$$q_k = (1-\alpha_q)q_{k-1} + \alpha_q p_{est}, \quad (5)$$

where q_k and q_{k-1} are the current and previous target model, $\alpha_q \in [0, 1]$ is the learning rate that contribute to the updated histogram and p_{est} is histogram of the estimated state, respectively. In order to determine when the estimate is reliable, the likelihood of the current estimation π_{est} is computed. The target appearance is then updated when this value is higher than an indicator of the expected likelihood value and is calculated with the following adaptive rule

$$\lambda_k = (1-\alpha_u)\lambda_{k-1} + \alpha_u \pi_{est}, \qquad (6)$$

here, λ_k is expected likelihood, $\alpha_u \in [0, 1]$ is the learning rate and π_{est} is estimated likelihood, respectively. This value indicates that the object has to be well matched to the model histogram before the update is applied.

2.6 Occlusion Prediction

In multiple objects tracking, the problem of occlusions can cause a failure in the tracking process. They may cause inaccurate in estimation and updating of the state of the tracked object since the likelihood measurement of the occluded target would be meaningless. Then, the resampling phase would propagate the wrong random samples and quickly cause the loosing of the object. Therefore, a proper handling of occlusions is very important task.

In this paper, the occlusions are predicted according to the dynamic models using the predicted distance between the objects and using likelihood measurement by comparing with recent historical values.

When the occlusion is detected, the object status turns into occluded object. This status involves several changes in the normal development of the process such as the adaptation rates of the state estimation $\alpha_{\mathbf{x}}, \alpha_{\dot{\mathbf{x}}}, \alpha_{\mathbf{w}}, \alpha_{\dot{\mathbf{w}}}$ are set to zero. It implies to the target estimated speed is kept constant and the estimated position is updated only according to its speed. In addition, no size or appearance adaptation is performed. Finally, those samples belonging to the occluded object are not resampled according to their weights but they are just propagated based on the transition model. As a result, samples spread around the occluded object, because of the uncertainty predictions terms. The other un-occluded object samples are normally resampled but they cannot be assigned to the occluded target. When the occlusion is no longer predicted or sample likelihood exceeds the value of previous likelihood of the occlusion object, the object status turns into not occluded object, which immediately implies the samples to be resampled again. In addition, position and speed are again updated.

3 Experimental Results

In order to evaluate our proposed method, we have performed the experiments to track the objects in the presence of occlusion in outdoor environment. The experiments are implemented on Pentium IV with 2.53 [GHz] CPU and 512 [MB] RAM. The resolution of each frame is 320×240 [pixels] image. Fig. 1 – Fig. 5 show the experimental results and data of the tracking objects. The small circle shows the estimated position of the tracking objects and the square shows the region of the tracked object used in color histogram calculation. On each experiment, we associate each target with individual template model and motion model.

On the first experiment, we tracked the multiple objects when they move in the opposite direction and occlude in the middle of the scene. In this experiment, the occlusion is detected based on likelihood measurement (Fig. 1 (a) and (b)). Based on the figure, we can see that the occluded object has a low likelihood compared with the recent historical value. It is occurred between frame #20 and frame #34. The tracking performance is shown in Fig. 2. Firstly, the first object appears on the left side of the scene and the second object appears afterward on the right side. After several frames, the first object occludes the second object in the middle of the scene (frame #20 – frame #34). The object is tracked successfully although partial occlusion (frame #20 and frame #34) and complete occlusions (frame #26) is occurred. The system successfully recovers the object from occlusion (frame #40). The occlusion is correctly handled by avoiding resampling of samples of the occluded object and appearance models updating. After the occlusion is no longer occurred, each object is normally resampled and the target appearance is update again.

On the second expriment, we implement our method to track the soccer players of the red team. The experimtal result is shown in Fig 3. In this experiment, the occlusion is not detected. Therefore, the resample step and update are performed normally on each target. We succesfully track the soccer players with our algorithm.

On the last experiment, we try to implement our algorihtm to track multiple objects in the complex environment. In this experiment, our targets are selected up to three objects. The complex occlusion occurs on each tracked object. The tracking performance is shown in Fig. 4. The object occlusion is firstly detected on frame #12 when the tracked object (in blue rectangle) occludes another object. In this condition, we can still measure the likelihood of the tracked object because the region of tracked object is not occluded. The second occlusion is detected on frame #45. In this condition, the likelihood cannot be measured and resample step is not performed because the tracked object (in yellow rectangle) is occluded by another object. We cannot extract the image information from the object. The samples are just propagated based on the transition model and spread around the object. Here, our method succesfully predict the target although it occludes by other object. The third occlusion is detected on frame #50. The occlusion occurs between the tracked objects. The third tracked object (in pink rectangle) occludes the second tracked object (in blue rectangle). In this conditon, we have to distinguish the samples associated on each object. The likelihood of the samples belong to occluded object (second object) is not measured and the resample step is not performed. However, the likelihood of the samples belong to the third object can still be measured and the resample step is performed normally. In this experiment, we succcefully handle the occlusion by redefnition of likelihood and resample process.

4 Conclusion

This paper presented a method to track multiple objects employing color-based particle filter in the presence of object occlusion. The color feature is utilized to evaluate the samples properly associated to the target. We rely on Bhattacharya coefficient between target and sample histogram to perform this task. The problem occlusion is handled by redefining the resample and target update of the samples. The occlusion itself is predicted based on the dynamical model and likelihood measurement. The performance of the tracking algorithm was evaluated by the experiments. The experimental results show the algorithm can successfully track multiple objects in the presence of occlusion.

However, from the experiments, we find that the processing time of the proposed method is related to the number of the objects to be tracked. It is because each object has individual template and moves independently. Thus, the particle filter should estimate each object based on their template and their motion model. Most of the time in our method is spent on building the color histogram whose complexity is proportional to the number of particles and size of the object regions. In our case, our implementation can run at 15~16 frames/s on two object tracking in average. However, in the football sequence and complex experiment, we can track three objects that run at 6~8 frames/s in average. The main drawback of the proposed method is that the increasing of number of objects makes the processing time become longer.

The performance of the objects tracking can be improved in several ways such as adding the background modeling information [10] in the calculation of likelihood, detection of appearing objects and disappearing already existed object using hybrid particle filter [13]. The further improvements could be performed to speed up the computational time and to obtain much better. In addition, performing the proposed method to realize multi-camera tracking is also interesting. Taking them into consideration could lead to some improvements. These are remaining for our future works.

References:

[1] T. Broida and R. Chellappa, “Estimation of object motion parameters from noisy images”, IEEE Trans. on Patt. Anal. and Machine Intell., vol. 8, no. 1, pp. 90–99, 1986.

[2] S. Arulampalam, S. Maskell, N. Gordon and T. Clapp, “A tutorial on particle filters for on-line nonlinear/non-Gaussian Bayesian tracking”, IEEE Trans. on Signal Processing, pp. 174–188, 2002.

[3] M. Isard and A. Blake, “Condensation: Conditional density propagation for visual tracking”, Int. J of Computer Vision, vol. 29, no. 1, pp. 5–28, 1998.

[4] G. Kitagawa, “Monte Carlo filter and smoother for non-Gaussian nonlinear state space models”, J. of Comput and Graph Stat, vol. 5, no. 1, pp. 1–25, 1996.

[5] N. Gordon, D. Salmon and A. Smith, “Novel approach to nonlinear/non-Gaussian Bayesian state estimation”, IEE Proc. on Radar and Signal Processing, vol. 140, no. 2, pp. 107–113, 1993.

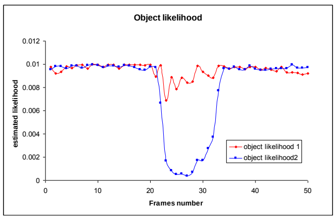
[6] B. Ristic, S. Arulampalam, and N. Gordon, “Beyond the Kalman filter: particle filters for tracking applications”, Artech House, 2004.

[7] H. Tanizaki, “Non-Gaussian state-space modeling of non-stationary time series, J. of American Statistic Association, vol. 82, pp. 1032–1063, 1987.

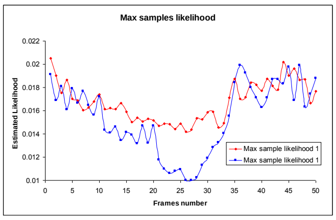
[8] O. King and D. Forsyth, “How does Condensation behave with a finite number of samples”, Proc. of the 6th European Conference on Computer Vision, pp. 695–709, 2000.

[9] K. Nummiaro , E. Koller and L. Gool, “An adaptive color-based particle filter”, Image and Vision Computing, vol. 25, no. 1, pp. 99–110, 2002.

[10] P. Perez, C. Hue, J. Vermaak and M. Gangnet, "Color-based probabilistic tracking", Proceedings of the 7th European Conference on Computer Vision, London, pp. 661–675, 2002.
[11] D. Comaniciu, V. Ramesh and P. Meer, "Kernel-based object tracking", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 25, no. 5, pp. 564–577, 2003.
[12] J. Deutscher and I. Reid, "Articulated body motion capture by stochastic search", Int J of Computer Vision, vol. 61, no. 2, pp. 185-205, 2005.
[13] J. Czyz, B. Ristic and B. Macq, "A particle filter for joint detection and tracking of color objects", Image and Vision Computing, vol. 25, no. 8, pp. 1271–1281, 2007.
[14] M. Isard and A. Blake, "A mixed-state condensation tracker with automatic model-switching", Proc. of the Int. Conference on Computer Vision, pp. 107–112, 1998.
[15] M. Isard, and J. MacCormick, "Bramble: a Bayesian multiple blob tracker", Proc. of Int. Conference on Computer Vision, pp. 34–41, 2001.
[16] B. Ristic, S. Arulampalam and N. Gordon, "Beyond the Kalman filter: Particle filters for tracking applications", Artech House, 2004.

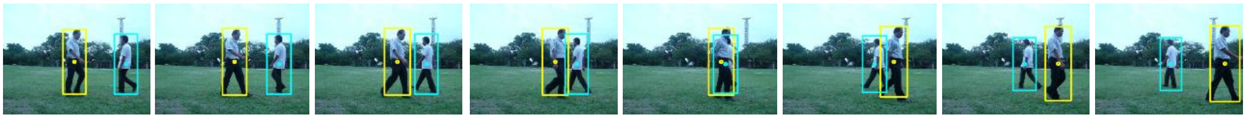


(a)



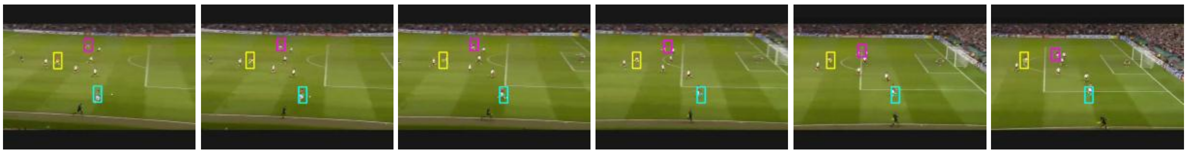
(b)

Figure 1 Likelihood measurement; (a) Estimated object likelihood, (b) Maximum object likelihood



#5 #10 #15 #20 #26 #34 #40 #50

Figure 2 Multiple objects tracking in the presence of occlusion



#1 #10 #20 #30 #40 #50

Figure 3 Tracking the soccer players

#1 #12 #40 #45 #50 #75 #80 #90

Figure 4 Multiple objects trackign in the complex environment

ISBN: 978-1-61804-062-6