MVA2005 IAPR Conference on Machine VIsion Applications, May 16-18, 2005 Tsukuba Science City, Japan

8-18

Particle Filter for Visual Tracking Using Multiple Cameras

Ya-Dong Wang, Jian-Kang Wu
Institute for Infocomm Research
21 Heng Mui Keng Terrace
Singapore 119613
{stuwy, jiankang}@i2r.a-star.edu.sg

Ashraf A. Kassim
Dep. of Electrical and Computer Engineering
National University of Singapore
Singapore 119260
ashraf@nus.edu.sg

Abstract

This article proposes an approach for visual tracking using multiple cameras with overlapping fields of view. A spatial and temporal recursive Bayesian filtering approach using particle filter is proposed to fuse image sequences of multiple cameras to optimally estimate the state of the system, i.e., the target's location. An approximation method for importance sampling function and weight update function is also proposed. Our results show that our algorithm is effective when complete occlusions occur. This method can be used for data fusion for multiple measurements in dynamic systems.

1 Introduction

Occlusion is extremely difficult for a single camera system. Having realized the limitation of using a single camera to track occluded objects, there is a trend to use multiple cameras. In this article, we focus on tracking moving objects using multiple cameras with overlapping fields of view in order to solve occlusion problem. We find that occlusions in one camera may be differentiated in the field of view of another camera and fusing data from multiple cameras at different locations can deal with occlusions in a particular camera. We also find that particle filter can automatically decide which particles (hypothesis) are close to the true state by weight updates and resampling. This property can be used for data fusion of multiple cameras and occlusion problems in a particular camera.

1.1 Previous work

The particle filter is a sequential Monte Carlo filter and can be applied to solve nonlinear and non-Gaussian problem [1]. Gordon et al. [2] proposed the *bootstrap* filter to implement the recursive Bayesian filter for nonlinear or non-Gaussian state estimation. The required probability density of the state vector is represented as a set of random samples, which are updated and propagated. The state vector x_t evolves according to the state model

$$x_t = f_t(x_{t-1}, w_t) \tag{1}$$

where f_t is the state transition function and w_t is a zero mean, white-noise sequence. The measurement y_t is related to the state vector via the observation equation

$$y_t = h_t(x_t, v_t) \tag{2}$$

where h_t is the measurement function and v_t is a zero mean, white-noise sequence. The available information at time step t is the set of measurements $Y_t=\{y_i\text{: } i=1, ..., t\}$. The objective is to construct the probability density function of the current state vector x_t, given all available information: $p(x_t|Y_t)$. In principle the Bayesian filtering consists of two stages: prediction and update. The prediction stage is

$$p(x_t \mid Y_{t-1}) = \int p(x_t \mid x_{t-1}) p(x_{t-1} \mid Y_{1:t-1}) dx_{t-1} \tag{3}$$

while the update stage is

$$p(x_t \mid Y_{1:t}) = \frac{p(y_t \mid x_t) p(x_t \mid Y_{1:t-1})}{p(y_t \mid Y_{1:t-1})} \tag{4}$$

Liu and Chen [3] presented a general sequential importance sampling framework for using Monte Carlo methods to dynamic systems. Doucet [4] provided an overview of sequential simulation based methods for Bayesian filtering of nonlinear and non-Gaussian dynamic models.

The *condensation* algorithm [5] used "factored sampling" to represent the probability distribution of possible interpretations by a randomly generated sample set. The learned dynamical models propagate the random sample set over time together with visual observations. Isard and Black [6] combined the statistical technique of important sampling with the condensation algorithm. The general framework is described as *Icondensation* and is demonstrated by a hand tracker which combines color blob-tracking with a contour model. Okuma et al. [7] combined the strengths of two successful algorithms: *particle filter* and *Adaboost* to produce a mixed importance sampling function.

An advantage of particle filter is to allow measurements from the various sensors to be fused in the Bayesian framework even when no knowledge is available about their dependence. This point is particularly useful for multiple cameras tracking. Perez et al. [8] introduce generic importance sampling mechanisms for data fusion and demonstrate the algorithm by fusing color with stereo sound or motion. Each of the three cues can be modeled by an appropriate data likelihood function and the intermittent cues (sound or motion) are best handled by generating proposal distributions from their likelihood functions. Their method is applied in the fusion of multiple cues (color and motion) and different types of sensors (camera and microphone).

1.2 Contributions

In this article, we use multiple cameras to track a moving object. Visual information obtained from cameras at different locations is fused to deal with the occlusion problem in a particular camera. Two main contributions in this article are: 1) a spatial and temporal recursive Bayesian filtering approach using Particle Filter for fusing multiple cameras' observations; 2) an approximation approach for the optimal importance sampling function and weight update function.

2 A Spatial and Temporal Recursive Bayesian Filtering Using Particle Filter for Multiple Cameras

There are some assumptions of our algorithm: 1) multiple cameras must have a common field of view; 2) only one target is tracked; 3) The target is detected in the first frame and the target model is known and constant during tracking. The last two assumptions can be removed by data association and adaptive appearance model in future work. At present we focus on data fusion of one target from multiple cameras. The limitation of our algorithm is: when occlusions occur, at least one camera can see the tracked target.

We propose a spatial and temporal recursive Bayesian filtering framework using particle filtering for fusing multiple observations from different locations and times. The fusion includes two types of information: 1) the fusion of observations from multiple locations at same time; 2) the fusion of the state of the previous time and observations of the current time.

C is the number of cameras used. x_t is the state of the system (e.g. the position and velocity of the moving object) and y_t^c is the measurement from the c^{th} camera at the time *t*. $Y_t^{1:C}$ are measurements from the *1st* camera to the c^{th} camera at the time *t*. $Y_{1:t}^{1:C}$ are sets of measurement of $Y_t^{1:C}$ from the time 1 to the time *t*, i.e.,

$$Y_{1:t}^{1:C} = \{y_1^1,\ldots,y_1^C, y_2^1,\ldots,y_2^C,\cdots,y_t^1,\ldots y_t^C\}$$

Therefore, the tracking problem is reduced to an inference problem and our objective is to construct the conditional probability

$$p(x_t \mid Y_{1:t}^{1:C}) \tag{5}$$

Similar to (4), the update stage is

$$p(x_t \mid Y_{1:t}^{1:C}) = \frac{p(Y_t^{1:C} \mid x_t) p(x_t \mid Y_{1:t-1}^{1:C})}{p(Y_t^{1:C} \mid Y_{1:t-1}^{1:C})}$$

$$\propto p(Y_t^{1:C} \mid x_t) p(x_t \mid Y_{1:t-1}^{1:C}) \tag{6}$$

We assume that the measurements are independent of each other given the state x_t because these measurements come from different cameras. Then (6) can be represented as:

$$p(x_t \mid Y_{1:t}^{1:C}) \propto \prod_{i=1}^{C} p(y_t^i \mid x_t) p(x_t \mid Y_{1:t-1}^{1:C}) \tag{7}$$

Similar to (3), the state prediction stage is

$$p(x_t \mid Y_{t-1}^{1:C}) = \int p(x_t \mid x_{t-1}) p(x_{t-1} \mid Y_{1:t-1}^{1:C}) dx_{t-1} \tag{8}$$

In [6], the optimal importance sampling function is $p(x_t \mid x_{t-1}^{(n)}, y_t)$ and the weight update function is $w_t^{(n)} = w_{t-1}^{(n)} p(y_t \mid x_{t-1}^{(n)})$. We propose that the importance function for multiple cameras is

$$p(x_t \mid x_{t-1}^{(n)}, Y_t^{1:C}) \tag{9}$$

Then the important weight is updated as

$$w_t^{(n)} = w_{t-1}^{(n)} p(Y_t^{1:C} \mid x_{t-1}^{(n)})$$

$$= w_{t-1}^{(n)} \prod_{i=1}^{C} p(y_t^i \mid x_{t-1}^{(n)}) \tag{10}$$

There are two problems in the above algorithm. First, in the most situations, (9) is not available in a closed-form. Second, (10) is easily affected by occlusions in a particular camera. We propose an approximation approach of our spatial temporal recursive Bayesian filtering algorithm. The importance sampling function is approximated as:

$$\alpha_0 p(x_t \mid x_{t-1}^{(n)}) + \sum_{i=1}^{C} \alpha_i p(x_t \mid y_t^i) \tag{11}$$

where $\sum_{i=0}^{C} \alpha_i = 1$ and the weight update function is approximated as

$$w_t^{(n)} = w_{t-1}^{(n)} \max_i p(y_t^i \mid x_t^{(n)}) \tag{12}$$

In our visual tracking task, x_t is the state to be estimated, i.e., the moving object's location. The observation from the c^{th} camera is a region

$$x = (s, t, width, height)$$

where (*s, t*) is the coordinate of the top left corner of the bounding box of the moving object in image coordinate and *width* and *height* are respectively the width and height of the bounding box of the moving object in image coordinate. The system dynamic equation is assumed as a constant position model,

$$x_t = x_{t-1} + w_t \tag{13}$$

where w_t is a zero mean, Gaussian white noise. The observation equation

$$y_t = x_t + v_t \tag{14}$$

where v_t is a zero mean, Gaussian white noise.

We use the 16× 16× 16 bins color histogram as the metric to update the weights of particles. The likelihood function is similar with the Bhattacharyya coefficient using in the *mean shift* algorithm [9]. The target model is a color histogram $\hat{q}_u$ where u is the color index. The candidate region of the n^{th} particles is the bounding box defined by the n^{th} particles. The color histogram of the candidate $x_t^{(n)}$ is $\hat{p}_u(x_t^{(n)})$. The color similarity metric is

$$\hat{\rho}(y)=\sum_{u=1}^{m}\sqrt{\hat{p}_u(x_t^{(n)})\hat{q}_u} \tag{15}$$

where m is the number of bins.

The coordinates of different cameras need be transformed into a common coordinate. We use a simple affine transformation to transform the coordinate of one camera into the coordinate of another camera, as follows:

$$\begin{bmatrix} x' \\ y' \end{bmatrix}=\begin{bmatrix} a & b \\ c & d \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix}+\begin{bmatrix} t_x \\ t_y \end{bmatrix} \tag{16}$$

where (x',y') and (x,y) are respectively the image coordinate of the first and second camera and a, b, c, d, t_x, t_y are transformation parameters.

The approximation approach of the spatial and temporal recursive Bayesian filtering for visual tracking and data fusion of multiple cameras is summarized in Figure 1.

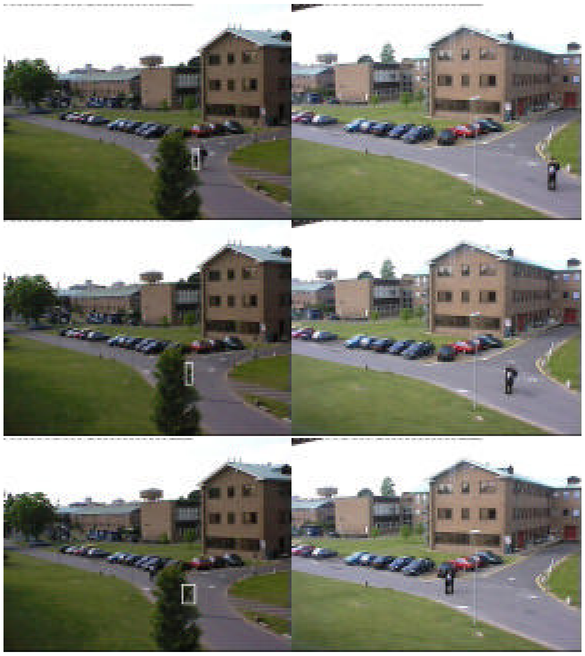
```
1. Initialize the target model $\hat{q}_u$. Time $t = 1$.
2. For $i = 1,\ldots,C$, use condensation algorithm to get the observation $\hat{y}_t^i$ of the $i$ camera. Transform each observation $\hat{y}_t^i$ into a common coordinate $y_t^i$.
3. For $n = 1,\ldots,N$, sampling $\tilde{x}_t^{(n)}$ from
       $\alpha_0 p(x_t \mid x_{t-1}^{(n)})+\sum_{i=1}^{C}\alpha_i p(x_t \mid y_t^i)$
4. For $n = 1,\ldots,N$, evaluate the importance weights
       $\tilde{w}_t^{(n)} = w_{t-1}^{(n)} \max_i p(y_t^i \mid x_t^{(n)})$
5. For $n = 1,\ldots,N$, normalize $\tilde{w}_t^{(n)}$
       $w_t^{(n)} = \frac{\tilde{w}_t^{(n)}}{\sum_{n=1}^{N}\tilde{w}_t^{(n)}}$   (17)
6. Resampling: For n = $1,\ldots N$, sample an index $j(n)$ distribution according to discrete distribution with $N$ elements satisfying $P(j(n)=l)=w_t^{(l)}$, then $x_t^{(n)}=\tilde{x}_t^{(n)}$ and $w_t^{(n)}=1/N$.
7. The current location is
       $E(x_t)=\frac{1}{N}\sum_{n=1}^{N}x_t^{(n)}$   (18)
8. $t=t+1$. Repeat step 2-7.
```

Figure 1. An approximation algorithm of the spatial and temporal recursive Bayesian filtering for visual tracking of multiple cameras

3 Results

Our method described above is tested using the PETS2001 data sequences [10]. The second PETS2001 datasets have two image sequences of two cameras from different views. There are five landmarks in the PETS2001 data where the three main paths meet. They are used to obtain the six parameters of the affine transformation. The tracking algorithm is implemented in Matlab. The number of particles is 50 in our experiments.

The results using condensation for frame 293, 352, 465 are shown in Figure 2. The target model is initialized by manual selecting a target region. Figure 2(a) are the tracking results using only observations from the first camera while Figure 2(b) are the tracking results using only observations from the second camera. As for the first camera, the human is total occluded by the tree and the tracking result is lost in frame 352. When the object appears again in frame 465, condensation can not track it. As for the second camera, condensation can accurately track the target. The measurements of the second camera provide more accurate observations than the measurements of the first camera when occlusions occur in the first camera.



(a) (b)

Figure 2. Tracking results using condensation for frame 293, 352 and 465

The experimental results using our data fusion algorithm for frame 293, 352, 465 are shown in Figure 3. We assume a uniform distribution for the weights of the importance sampling function in (11). Therefore, $\alpha_0=\alpha_1=\alpha_2=1/3$. Figure 3 (a) and (b) show data fusion results using observations of both the first and second cameras. For frame 352, although some observations of the first camera are wrong, our tracking algorithm is still able to track the person in frame 352 using the observations of the other camera. When the person appears again in frame 465, our algorithm can localize it. The results show that multiple cameras tracking and data fusion using particle filter can fuse spatial and temporal measurements of multiple cameras and solve the occlusion problem in a particular camera. The trajectory of the occluded target can be recovered according to the observations of another camera.

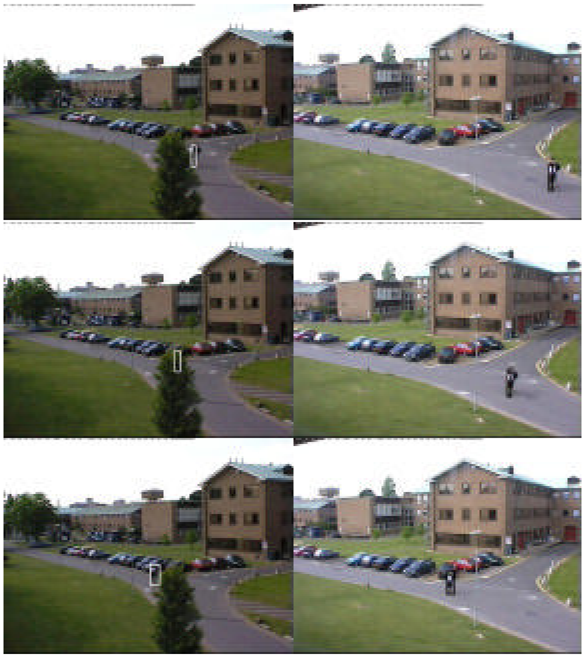


Figure 3. Tracking and data fusion results using particle filter for frame 293, 352 and 465

4 Conclusion

This article proposed an approach for visual tracking using multiple cameras with overlapping fields of view. A spatial and temporal recursive Bayesian filtering using particle filter for fusing multiple measurements was presented. The approximation approach in the implementation showed that our approach can automatically recover the trajectory of the completely occluded target. There are many future works. We are going to relax the assumptions of our algorithm so that we can track multiple targets and use adaptive appearance model. We will also extend our work to deal with multiple cues in an image such as contour, motion and color etc. and multiple modalities of sensors such as camera, ultrasonic, infrared and microphone etc. to achieve reliable tracking in various conditions.

Refe rences

[1] A. Doucet, N. de Freitas, and N.J. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Series Statistics for Engineering and Information Science. New York: Springer-Verlag, May 2001.

[2] N.J. Gordon, D.J. Salmond, and A.F.M. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. *Proceedings IEE. F*, 1400(2):107-113, 1993.

[3] J.S. Liu and R. Chen. Sequential monte carlo methods for dynamic systems. *Journal of the American Statistical Association*, 93:1032-1044, 1998.

[4] A. Doucet, S. Godsill, and C. Andrieu, On sequential Monte Carlo sampling methods for Bayesian filtering, *Statistics and Computing*, 10(3):197-208, 2000.

[5] M. Isard and A. Blake. Condensation-conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5-28, 1998.

[6] M. Isard and A. Blake. Icondensation: Unifying low-level and high-level tracking in a stochastic framework. In *Proceedings of 5th European Conference Computer Vision*, volume 1, pages 893–908, 1998.

[7] K. Okuma, A. Taleghani, N. de Freitas, J.J. Little, and D.G. Lowe. A boosted particle filter: Multitarget detection and tracking. In *Proceedings of 8th European Conference Computer Vision*, pages 28-39, 2004.

[8] P. Perez, J. Vermaak, and A. Blake. Data fusion for visual tracking with particles. *Proceedings of the IEEE*, 92(3):495–513, 2004.

[9] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. In *Proceeding of IEEE Conference Computer Vision and Pattern Recognition*, 2000.

[10] http://www.visualsurveillance.org/.