2010 International Conference on Electrical and Control Engineering

Pedestrian Tracking Using Particle Filter Algorithm

Xu Fen, Gao Ming

College of Mech-electrical Engineering,

North China University of Technology

Beijing ,China,100144

Abstract—**Pedestrian tracking is a difficult task due to the complexity of environment and the irregular motion of human body. Particle Filters are advantageous on solving nonlinear problems with non-gaussian system noise. By extracting the target color-histogram features and calculating the similarity between particle candidates and target template region through discrete Bhattacharyya Coefficient, this paper presents a particle filter algorithm for pedestrian tracking. Experimental results show that the proposed algorithm outperforms Kalman tracking in almost all situations, especially when the target is occluded by other objects.**

Keywords- Pedestrian Tracking; Particle Filter; Color Histogram; Bhattacharyya Coefficient

I. INTRODUCTION

Moving target tracking is an important application area of computer vision. The important attributes of a good algorithm for visual tracking problems include tracking accuracy, tracking speed and robustness. Different tracking methods have been developed, such as region-based tracking, feature-based tracking, deformable-template-based tracking, model-based tracking, and among others. For human body tracking three types of models, namely, line graph model, 2D model and 3D models, are usually used. Tracking algorithm of mathematical tools commonly used Kalman Filter, Particle Filter, DBN (Dynamic Bayesian Network) and so on.[1]-[4]

Moving human tracking, or pedestrian tracking, is a special tracking problem with more challenge. The task of human tracking is to establish the link of various moving objects among the frames via certain features, such as position, velocity, shape, texture, color, etc. The human body is non-rigid, and structurally complex. The body's movement has a great freedom and a high degree of nonlinearity. Besides, the human motion could be partially hid or occluded by shelters or other human bodies, especially in the case of tracking with a group of people. For these reasons, particle Filter is considered quite suitable for this task.

This paper is organized as follows: Particle Filter is introduced in section Ⅱ. Color histogram - particle filter tracker is provided in section Ⅲ, followed by experiment results in section Ⅳ. Finally conclusion is given in section Ⅴ.

II. PARTICLE FILTER

The nature of particle filter is based on Monte Carlo method, i.e. using discrete particle sample-set to denote the probability density function. The key idea of particle filter is to approximate the probability distribution by a weighted sample set $S = \{(s^{(n)}, w^{(n)}) \mid n = 1 \dots N\}$, where s(n) indicates the nth particle, w(n) indicates the importance of the particle, or the weight. Particle filter is a sequential importance sampling (SIS) method, and can be used for any form of state-space models.

For a statistically stable random process, the posterior probability density function of the system state is described by $p(x_{t-1} \mid z_{t-1})$ at time t-1. According to certain rules, N samples are randomly selected and the observation vector z_t at time t is obtained. Usually, regions of high densities are expected to have more particles, or to have particles with greater weights. After a few iterations of state update, the

Beijing Municipal Educational Committee Sci. & Tech. Development Plan (KM200410009003)

978-0-7695-4031-3/10 $26.00 © 2010 IEEE
DOI 10.1109/iCECE.2010.364

posterior probability density function at time t can be approximated as $p(x_t \mid z_t)$.

III. COLOR HISTOGRAM BASED PARTICLE FILTER

A. System propagation model

The system propagation model describes the transfer form of target states over time, here for pedestrian tracking, the location change of target person. This step is also known as the propagation of particles or particle sampling.

State vector needs to be able to describe the human motion-related parameters, such as position, speed and so on. Particle is expressed as $\{x_{0:t}^i, w_t^j\}_{i=1}^N$. $x_{0:t}^i$ is the target state vector and can be expressed as

$$x_{0:t}^{i} = \{x_t^i, y_t^i, \dot{x}_t^i, \dot{y}_t^i\}_{t=0}^{t} \quad (1)$$

Where x_t^i, y_t^i specify the location of the target human, $\dot{x}_t^i, \dot{y}_t^i$ the motion, w_t^j is particle weight.

The sample set is propagated through the application of a dynamic model

$$x_t = x_{t-1} + v_t + w_t \quad (2)$$

v_t is the relative motion of target in each frame, w_t is a multivariate Gaussian random variable.

B. System observation model

As for system observation model, we first extract the color histogram feature of target. For visual tracking. color is considered a good robust feature, suitable to describe the deformation of targets, as well as the planar rotation of non-rigid objects. Common color features used include color histogram, color sets, color moments, color polymerization, and color-related maps. The track algorithm needs to compare two distributions: the color distribution of target template and the color distribution of each candidate region, i.e. particle. The color distribution of the target area is obtained by discretization of the color histogram. In our experiments, the histograms are typically calculated in the RGB space using 8×8×8 bins.

As the target area is a rectangle, it is bound to include non-target background color information. In order to reduce the influence of background color, we employ Kernel function

$$k(r) = \begin{cases} 1 - r^2 & r < 1 \\ 0 & otherwise \end{cases} \quad (3)$$

where r is the distance from the region center.

The color distribution $\hat{p}(x) = \{\hat{p}(x)\}_{u=1\cdots m}$ at location x is calculated as

$$\hat{p}_u(x) = C_h \sum_{i=1}^{n_h} k\left(\left\|\frac{x - x_i}{h}\right\|^2\right) \delta\left[b(x_i - u)\right] \quad (4)$$
$$u = 1, \cdots, m$$

where n_h is the number of pixels in the region, δ is the Kronecker delta function, the parameter $h = \sqrt{H_x^2 + H_y^2}$ is used to adapt the size of the region, H_x, H_y the half-width and half-height of rectangular area and the normalization factor

$$C_h = \frac{1}{\sum_{i=1}^{n_h} k\left(\left\|\frac{x - x_i}{h}\right\|^2\right)} \quad (5)$$

Which ensures that $\sum_{u=1}^{m} \hat{p}_u(x) = 1$.

Secondly, we have to define the similar function. similar function describes the similarity between the color distribution of particle candidate and that of target template. Discrete two color distributions $p = \{\hat{p}_u(x)\}_{u=1\cdots m}$ and $q = \{\hat{q}_u\}_{u=1\cdots m}$, the Bhattacharyya coefficient of two vectors is defined as

$$\rho\left[\hat{p}(x),\hat{q}\right]=\sum_{u=1}^{m}\sqrt{\hat{p}_u(x)\hat{q}_u} \qquad (6)$$

From the geometrical point of view, Bhattacharyya coefficient means the cosine of angle between two m-dimensional vectors. A greater Bhattacharyya coefficient indicates a smaller angle between the two vectors, hence a greater similarity between two vectors.

Similarity function is defined as

$$d_j\left(\hat{p}(x),\hat{q}\right)=\sqrt{1-\rho\left[\hat{p}(x),\hat{q}\right]} \qquad (7)$$

which is called the Bhattacharyya distance.

Observation probability density function is defined as:

$$p\left(z_t \mid x_t^j\right)=\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{1}{2\sigma^2}d_j^2} \qquad (8)$$

Sequence of the particle weight update formula is as followed:

$$w_t^j = w_{t-1}^j p\left(z_t \mid x_{t-1}^j\right) \qquad (9)$$

After the calculation of weighted particles, use the minimum variance estimation method to determine the target location

$$x_{new}=\sum_{i=1}^{N_s} x_t^i w_t^i \qquad (10)$$

C. Re-sampling

During propagation, particles are prone to degeneracy, i.e. a very small number of particles have considerable large weight, while all the other particles have negligible weights. In this case, re-sampling will be carried out as followed:

(1) Calculate the sum of i normalized particle weights $\sum_{n=1}^{i} w^i \in (0,1)$

(2) Generate a uniformly distributed random number $rand \in (0,1)$

(3) Find the smallest number j for which $\sum w^j \geq rand$

(4) Set $x^i = x^j, y^i = y^j$

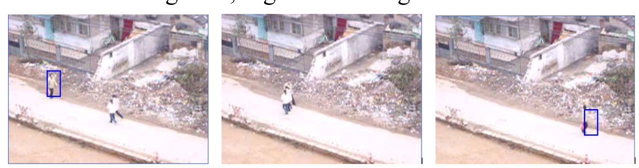
D. Update of target template

In the tracking process, due to the change of light and/or perspective, the target color distribution could vary to certain extent. To improve the robustness of tracking, the target template is updated timely. By introducing a factor α, the color information of the target in the past is dropped from frame to frame, The update of the target template is implemented by the following equation.

$$M_{updated}=\alpha\cdot M_{fixed}+(1-\alpha)M_{new} \qquad (11)$$

IV. REALIZATION AND EXPERIMENTAL RESULTS

The algorithm is implemented in Visual C++ and tested on a Pentium IV 2.8G 512M computer. Three videos are used for test. These videos cover situations of targets with translation or rotation movement, existence of occlusion, and/or similar objects, as well as the illumination changes. The tracking speed is around 50 frames / sec by estimation. Test results are shown in Figure 1, Figure 2 and Figure 3.



(a) frame 87 frame 136 frame 293

(b) frame 87 frame 136 frame 293

Figure 1 (a) Kalman filter tracking (b) Particle filter tracking.

For test video 1, with Kalman-filter based tracking, as the block in front of which the target stood is quite similar to the human body in term of colors，processing time is long, tracking failed with frame 136, and tracking accuracy is not good with frame 293. With the particle-filter based tracking,, the processing time is shorter, for the algorithm has good adaptability with object occlusion. Target is successfully tracked through the video..

(a) frame 40 frame 82 frame 184

(b) frame 40 frame 82 frame 184

Figure 2 (a) Kalman filter tracking (b) Particle filter tracking.

For test video 2, with the Kalman-filter based algorithm, the locations of target in frame 82 and 184 are not very accurate, possibly due to the trees of background and the turning of human body. With the particle-filter based tracking, however, the target is successfully tracked through the video.

(a) frame 1 frame 16 frame 45

(b) frame 1 frame 16 frame 45

Figure 3 (a) Kalman filter tracking (b) Particle filter tracking.

For test video 3, with Kalman filter based tracking, tracking failed with frame 45, due to the occlusion of target by similar objects. With particle filter based tracking, the target is successfully tracked through the video.

The trajectories of pedestrians tracked in the three videos are shown in Figure 4. Horizontal axis tells the x coordinate, while vertical axis gives the y coordinate. The unit is pixel.

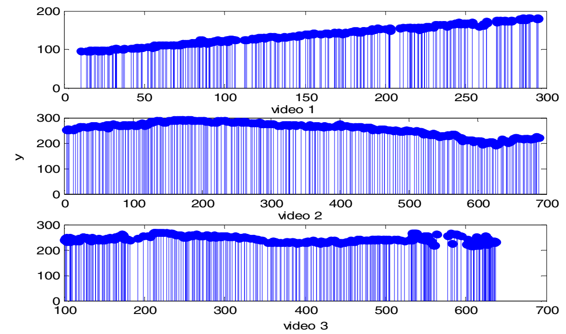


Figure 4 Trajectory of tracked target in the videos

V. CONCLUSION

The presented pedestrian tracking method based on particle filter can effectively track non-rigid and fast moving human targets. In comparison to Kalman filter based tracking, algorithm, the particle filter method has a better performance especially when target is sometimes occluded by other objects during the tracking. Problems also exist, e.g. the color histogram feature is sometimes not sufficient when dealing with a crowd of people. Future work is planned on creating joint-features state vector for pedestrian tracking, e.g. taking into account of the moment of motion in two directions.

[1] Meyer F, Bouthemy P. Region-based tracking using a fine motion models in long image sequences. CVCIP: Image understanding,1994,60(2):119-140

[2] Tissainayagam P, Suter D. Object tracking in image sequences using point feature. Pattern Recongnition,2005,38(1)105-113

[3] Yu Z, Jain A K, Dubuisson-Jolly M. Object tracking using deformable templates. IEEE Transactions on Pattern Analysis and Machine Intelligence,2000,22(5):544-549

[4] Hu W M, Tan T N, Wang L, Maybank S. A survey on visual Surveillance of object motion and behaviors. IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews 2004,34(3):334-352

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

[6] K. Nummiaro, E. Koller-Meier, L. Van Gool. An adaptive color-based particle filter. Image and Vision Computing, 2003, 21(1): 99-110.

[7] Huimin QIAN, Yaobin MAO, Jason GENG, Zhiquan WANG. Object tracking with self-updating tracking window. PAISI 2007.