PHD Filter for Multi-target Visual Tracking with Trajectory Recognition

Jingjing Wu
School of Aeronautics and Astronautics
Shanghai Jiao Tong University
Shanghai, China
wjjlady720@sjtu.edu.cn

Shiqiang Hu
School of Aeronautics and Astronautics
Shanghai Jiao Tong University
Shanghai, China
sqhu@sjtu.edu.cn

Abstract-Probability hypothesis density (PHD) filter, as a multi-target recursive Bayes filter, has generated substantial interest in the visual tracking field due to its ability to handle a time-varying number of nonlinear targets. But the target's trajectory cannot be identified within its own framework. To complement the ability of PHD, the auction algorithm is combined to calculate the object trajectories automatically. We present the motion detection, dynamic and measurement equation, as well as visual multi-target tracking algorithm based on Gaussian mixture probability hypothesis density (GM-PHD) in details. Experimental results on a large video surveillance dataset show the proposed multi-target tracking framework improves the tracker and recognizes the tracks when a variable number of targets appear, merge, split and disappear even in cluttered scenes.

Key Words: Probability hypothesis density (PHD) filter, Random finite set (RFS), Gaussian mixture, Auction.

1 Introduction

Multi-target tracking (MTT) is devoted to estimating the number of targets and their current states (position, velocity etc) based on a set of uncertain observations[1] (i.e. measurements).These observations are usually uncertain and inaccurate because of frequent occlusions among targets, arrival and departure of targets from surveillance region, as well as detection errors and clutters etc. Tracking a time varying number of targets is one of challenging tasks in multi-target visual tracking field due to the uncertainty between the observed detections and tracks, as well as the inaccuracy of observations.

In order to deal with inaccuracy originated from clutters, Kalman[2] and Particle filters[3] for single-target tracking are brought forward. The extension of these algorithms to multi-target tracking can be implemented by respectively applying single-target filtering after associating the measurements to the confirmed tracks of targets. Multiple hypothesis tracker[4] (MHT) proposed by Reid and joint probabilistic data association filter[1] (JPDAF) proposed by Bar-Shalom are two popular and effective algorithms for establishing the correspondence between targets and observations, which have been applied to track pedestrians[5] and motorcycles[6]etc. MHT, a measurement-oriented approach, assumes that each measurement originates either from a known target, a new target, or clutter, which requires an exponential amount of computations to enumerate all the possible hypotheses. JPDAF algorithm computes the association probabilities of the given measurements to various targets in validation gate and sums these measurements weighted by association probabilities as hypotheses on new observations.

An emerging multi-target tracking approach based on random finite set (RFS) has aroused increasing attention for its ability of estimating states of multiple targets avoiding associating the observations with targets. Under the RFS framework, the set of states of various targets and observations with the time evolution are modelled as RFSs, while the problem of dynamically estimating states of multiple targets immersed in clutter is formulated as Bayesian filtering[7]. However, the dimensionality of the RFS still grows with the number of targets, thus making the propagation of the full posterior density intractable. Mahler proposed a less computationally intensive alternative to propagate the probability hypothesis density (PHD), which is the first moment of the multi-target posterior density. The PHD recursion will get closed-solution by assuming the PHD to be a mixture of Gaussians, while the integrals involved in PHD recursion can also be approximated by a generic sequential Monte Carlo technique, i.e., GM-PHD[8] and Particle-PHD[9]respectively. Due to the merits on handling a time-varying number of nonlinear targets, the latter approach has been used in visual tracking systems [10]. Whereas the reliability of extracting state estimates from the particles representing the posterior intensity depends on clustering techniques, and the number of particles increasing with the number of targets will lead to higher computation cost. Besides, since the estimates from PHD filter are also RFS, PHD filter is not able to provide any information on the identity of targets which is indispensable in many video analysis tasks.

In this work, we propose a multi-target tracking framework with the ability of estimating the positions and scales for visual targets and recognizing their tracks in real-world scenarios. The observations (i.e., positions and sizes of objects) are computed from video using a modified subtraction detector. Next, the GM-PHD filter is adapted to reduce clutter and estimate the states and number of targets appearing, merging, splitting and disappearing in the surveillance region. Finally, the state estimates are associated with target tracks using the proposed approach based on auction algorithm. The block diagram of the proposed multi-target visual tracking method is presented in fig.1.

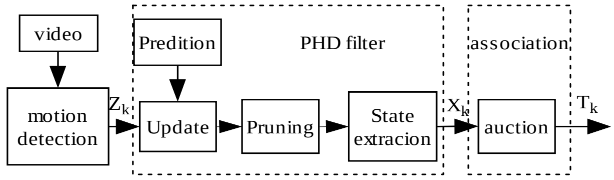


Fig.1 Visual tracking scheme of the proposed tracker

2 PHD filter

Let us represent the state of a target by a state vector x, then, in a multi-target scenario, a state set of multiple targets with variable number $M(k)$ at time k is random finite set $X_k = \{x_{k,1}, \cdots, x_{k,M(k)}\}$. As to the sensor, let $N(k)$ observations at time k be represented as random finite set $Z_k = \{z_{k,1}, \cdots, z_{k,N(k)}\}$. Analogous to the single-target Bayesian filtering, the uncertainty of the states and measurements in the random finite set formulation of multi-target filtering is modelled as RFSs, i.e., X_k and Z_k respectively. Furthermore, the dynamics in the states evolution and the randomness in the observations are depicted by the multi-target transition density $f_{k|k-1}(X_k \mid X_{k-1})$ and likelihood $p(Z_k \mid X_k)$. Then, the optimal multi-target Bayes filter in the RFS formulation can be obtained by propagating the multi-target posterior density with the time evolution. Nevertheless, multiple integrals involved in the recursion of multiple-target posterior are computationally intractable. To alleviate the computation complexity, the PHD filter is proposed.

Instead of propagating posterior density on multi-target state space, the PHD filter propagates a first-order statistical moment of the posterior multi-target state[7], which is defined as the intensity with its integral on any region of the state space giving the number estimate of targets in it. Moreover, the local maxima of PHD indicate the highest local concentration of the expected number of elements. Taking advantage of this character of PHD, the likely position of targets can also be obtained.

Accounting for target motion, birth and death in time, states in a given time are modelled as state RFSs, i.e., state RFS of the surviving targets from the previous time, state RFS of new targets and RFS of targets spawned from targets at the previous time. Consequently, the intensity $v_k(x)$ at time k could be divided into 3 parts: intensity of RFS of the surviving targets from a target with previous state ζ modelled by the survival probability $p_{S,k}(\zeta)$ and $f_{k|k-1}(X_k \mid X_{k-1})$, intensity of the birth RFS $\gamma_k(\cdot)$, and intensity of RFS for spawned targets $\beta_{k|k-1}(\cdot \mid \zeta)$. Then the PHD recursion is formulated by prediction and update steps, whose equations are shown in (1) and (2) respectively, i.e.,

$$v_{k|k-1}(x) = \int p_{S,k}(\zeta) f_{k|k-1}(x \mid \zeta) d\zeta + \int \beta_{k|k-1}(x \mid \zeta) v_{k-1}(\zeta) d\zeta + \gamma_k(x) \quad (1)$$

where $v_{k|k-1}(x)$ denotes the multi-target predicted density.

$$v_k(x) = [1 - p_{D,k}(x)] v_{k|k-1}(x) + \sum_{z \in Z_k} \frac{p_{D,k}(x) p_k(z \mid x) v_{k|k-1}(x)}{\kappa_k(z) + \int p_{D,k}(\xi) p(z \mid \xi) v_{k|k-1}(\xi)} \quad (2)$$

where $p_{D,k}(x)$ denotes the detection probability of a state x at time k, and $\kappa_k(\cdot)$ indicates the intensity of the clutter RFS with Poisson distribution taking the form $\kappa_k(\cdot) = \lambda u(z)$. λ is the average number of clutter over the surveillance region(e.g., the field of view of a camera) and $u(z)$ is the probability density of clutter over the surveillance region.

3 Moving targets detection

The motion detection method in this paper consists of four main processes: thresholding, background update by a modified updating approach, noise cleaning and connected component analysis. It is easy to obtain the background image when the camera is fixed. So we average the intensity of multiple background images as initial background denoted by B_{ini}. Assume that the current image is I_k, thresholding formulation is given by

$$P_k = | I_k - B_k | = \begin{cases} 1, if\ d \geq Threshold \\ 0, if\ d < Threshold \end{cases} \quad (3)$$

where B_k is the background image at time k updated at the previous time $k-1$ (if $k=1$, then $B_k = B_{ini}$), and P_k is binary difference image between the current image and the current background. Therefore, all pixels that correspond to foreground objects have the value 1, and all the other pixels are taken as background with value 0. The value of threshold can be selected by adaptive techniques like the Otsu thresholding, or be a constant for a simple scenario.

The modified background update method is based on the principle that, instead of using the all the information in the current frame of the video sequence, only the background pixels should be employed to update the background image, for the foreground pixels in the current image may pollute the updated background image.

Therefore, we need to classify the pixels in the current image as foreground and background before update. We take the binary image P_k as binary mask to replace foreground pixels in the current image I_k with the background pixels in previous background B_k. The modified updating equation is as follow,

$$B_{k+1} = (1-\alpha)B_k + \alpha * P_k .* B_k + \alpha * (1-P_k) .* I_k \quad (4)$$

where α ($0 < \alpha < 1$) denotes the updating velocity of taking advantage of the current image and empirically determined to be 0.2. The operator $.*$ denotes element-by-element multiplication of two arrays with the same dimension.

Thresholding, however, is not enough to get clear foreground regions. Due to ingredients like illumination changes, a significant level of noise would still exist on the result image. Therefore, the close operation (i.e., an iteration of erosion and dilation) is employed to foreground pixels to eliminate noise points. Moreover, a connected component operator is applied to find and eliminate bigger noises. Finally, after the step of noise cleaning, the centriod and scale of likely foreground objects (i.e., blobs) are computed by a binary connected component analysis.

4 Multi-target tracking

4.1 Filtering detection results using GM-PHD filter

In this work, the target region in an image is approximated with a $w \times h$ rectangle. Let the state of a target be $x_k = (vel_{x,k}, pos_{x,k}, vel_{y,k}, pos_{y,k}, w, h)^T$, where $\boldsymbol{pos}_k = (pos_{x,k}, pos_{y,k})$ is the centriod and $\boldsymbol{vel}_k = (vel_{x,k}, vel_{y,k})$ is the target speed at time k. Let observation of a single target $z_k = (\boldsymbol{pos}_k, w, h)^T$ be obtained by the detector presented in part 3. Assume that each target follows a linear Gaussian dynamical model and the measurement model is linear Gaussian, i.e.,

$$x_k = Fx_{k-1} + v_k \quad (5)$$

$$z_k = Hx_k + w_k \quad (6)$$

where F is the state transition matrix, v_k is the zero-mean Gaussian white process noise with covariance Q_v, H is the measurement matrix, and w_k is the zero-mean Gaussian white observation noise with covariance Q_w. F, Q_v, H and Q_w are given below,

$$F = \begin{pmatrix} I_2 & A & 0_2 \\ 0_2 & I_2 & 0_2 \\ 0_2 & 0_2 & I_2 \end{pmatrix} \quad (7)$$

$$Q_v = \sigma_v^2 \begin{pmatrix} \frac{\Delta^4}{4} I_2 & \frac{\Delta^3}{2} I_2 & 0_2 \\ \frac{\Delta^3}{2} I_2 & \Delta^2 I_2 & 0_2 \\ 0_2 & 0_2 & \Delta \end{pmatrix} \quad (8)$$

$$H = \begin{pmatrix} I_2 & 0_2 & 0_2 \\ 0_2 & 0_2 & I_2 \end{pmatrix} \quad (9)$$

$$Q_w = \sigma_w^2 I_4 \quad (10)$$

where I_n and 0_n denote, respectively, the $n \times n$ identity and zero matrices, Δ is the sampling period, σ_v is the standard deviation of process noise, σ_w is the standard deviation of measurement noise, and $A = \begin{pmatrix} \Delta & 0 \\ 0 & \Delta \end{pmatrix}$. Let us approximate PHD of the birth RFS using $J_{\gamma,k+1}$ Gaussian mixtures with mean $\{m_{\gamma,k+1}^{(i)}\}_{i=1}^{J_{\gamma,k+1}}$ and covariance $\{p_{\gamma,k+1}^{(i)}\}_{i=1}^{J_{\gamma,k+1}}$ and associated weights $\{\omega_{\gamma,k+1}^{(i)}\}_{i=1}^{J_{\gamma,k+1}}$, and approximate PHD of the spawn RFS by $J_{\beta,k+1}$ Gaussian mixtures with mean $\{m_{\beta,k+1}^{(i)}\}_{i=1}^{J_{\beta,k+1}}$ and covariance $\{p_{\beta,k+1}^{(i)}\}_{i=1}^{J_{\beta,k+1}}$ and associated weights $\{\omega_{\beta,k+1}^{(i)}\}_{i=1}^{J_{\beta,k+1}}$ at time $k+1$. The main algorithm is summarized as follows.

At time $k \ge 0$, let Gaussian mixtures with parameters $\{\omega_k^{(i)}, m_k^{(i)}, p_k^{(i)}\}_{i=1}^{J_k}$ denote an approximation of PHD.

- Detection

Process the images using the modified motion detector described in section 3. The centriods and scales of all foreground blobs are obtained as the measurement set Z_{k+1} at time $k+1$.

- Prediction

For $j = 1, \cdots, J_{\gamma,k+1}$, predict PHD of birth RFS for time $k+1$ by substituting them with Gaussian mixtures approximation of the PHD of birth RFS at time k. For $j = 1, \cdots, J_{\beta,k+1}$ and $l = 1, \cdots, J_k$, predicting PHD of spawned RFS is given by

$$\upsilon_{\beta,k+1|k}^{(i)}(x) = \sum_{j=1}^{J_k} \sum_{l=1}^{J_{\beta,k+1}} \omega_k^{(j)} \omega_{\beta,k+1}^{(l)} \mathrm{N}(x; m_{\beta,k+1|k}^{(j,l)}, F_{\beta,k+1|k}^{(j,l)}) \quad (11)$$

$$m_{k+1|k}^{(i)} = d_{\beta,k}^{(j)} + F_{\beta,k}^{(j)} m_k^{(l)} \quad (12)$$

$$P_{k+1|k}^{(i)} = Q_{\beta,k}^{(j)} + F_{\beta,k}^{j} P_k^{(l)} (F_{\beta,k}^{j})^T \quad (13)$$

$$\omega_{k+1|k}^{(i)} = \omega_k^{(l)} \omega_{\beta,k+1}^{j} \quad (14)$$

For $l = 1, \cdots, J_k$, according to (5) predict PHD for existing targets

$$\omega_{k+1|k}^{(i)} = p_S \omega_k^{(j)} \quad (15)$$

$$m_{k+1|k}^{(i)} = F_k m_k^{(j)} \quad (16)$$

$$P_{k+1|k}^{(i)} = Q_k^{(j)} + F_k P_k^{(j)} (F_k)^T \quad (17)$$

where p_S is the survival probability.

- Update

For $j = 1, \cdots, J_{k+1|k}$ ($J_{k+1|k} = J_{\gamma,k+1} + J_k J_{\beta,k+1} + J_{k+1}$), compute the updated values for missed detections

$$\omega_{k+1|k}^{(j)} = (1 - p_D) \omega_{k+1|k}^{(j)} \quad (18)$$

$$m_{k+1}^{(j)} = m_{k+1|k}^{(j)} \quad (19)$$

$$P_{k+1}^{(j)} = P_{k+1|k}^{(j)} \quad (20)$$

For each $z \in Z_{k+1}$ and $j = 1, \cdots, J_{k+1|k}$, according to the likelihood defined in (6) and compute the updated weight, mean and covariance

$$\omega_{k+1}^{(j)}(z) = \frac{p_D \omega_{k+1|k}^{(j)} \mathrm{N}\,(z; Hm_{k|k+1}^{(j)}, R + HP_{k+1|k}^{j} H^T)}{\kappa_k(z) + p_D \sum_{l=1}^{J_{k+1|k}} \omega_{k+1|k}^{l} \mathrm{N}\,(z; Hm_{k|k+1}^{(l)}, R + HP_{k+1|k}^{l} H^T)} \quad (21)$$

$$m_{k+1}^{(j)} = m_{k+1|k}^{(j)} + K_{k+1}^{(j)} (z - Hm_{k+1|k}^{(j)}) \quad (22)$$

$$P_{k+1}^{(j)} = P_{k+1|k+1}^{(j)} \quad (23)$$

- Pruning

Discard Gaussian components with weights below some preset threshold and merge closely spaced Gaussian components using a single Gaussian. After the pruning step, we will get the Gaussian mixture approximation of PHD at time $k+1$, i.e., $\left\{ \omega_{k+1}^{(i)}, m_{k+1}^{(i)}, p_{k+1}^{(i)} \right\}_{i=1}^{J_{k+1}}$.

- State extraction

Select the means of the Gaussians with weights greater than a preset threshold, e.g., 0.5, as the states (i.e., position and scale) of multiple targets. Denote the final states as X_{k+1}.

4.2 Recognizing tracks for targets

Trajectory recognition is to obtain the consistent identity for targets over time. In this work, generating a track for a specific target is to assign the state of an individual target at time $k+1$ to its confirmed state or track at time k. The assignment is performed by designing a cost function which is based on point-matching of the estimated state (position and velocity) with the confirmed state at previous time. Other than the position information, we take into account velocity information to the cost function for it contains direction change of motion. The auction algorithm[12] based association method of this paper is designed as follow.

Step1: Compute the association matrix

Denote $N(k+1)$ and $L(k)$ are the element number of state set X_{k+1} after PHD filtering and confirmed state set Y_k at time k. If $N(k+1) >= L(k)$, for each $x_{k+1}^{i} = (\boldsymbol{pos}_{k+1}^{i}, \boldsymbol{vel}_{k+1}^{i}, w_{k+1}^{i}, h_{k+1}^{i})^T \in X_{k+1}$ and each confirmed state $y_k^j = (\boldsymbol{pos}_k^j, \boldsymbol{vel}_k^j)^T \in Y_k$ at time k, compute Euclid distance for position and velocity as the cost on assigning x_{k+1}^{i} to y_k^j using equation (24).

$$C_{i,j} = \sqrt{\sum_{1}^{a} (x_{k+1}^{i} - y_k^j)^2} \quad (24)$$

where a is the dimension of y_k^j, and only compute position and velocity entries in x_{k+1}^{i}. Then the $N(k+1) \times L(k)$ association matrix is obtained. If $N(k+1) < L(k)$, then reverse the association matrix.

Step2: Initialize all unassigned states in X_{k+1} and set the track prices to be equal to zero.

Step3: Select one unassigned state x_{k+1}^{m} and find the “best” track y_k^n for it according to equation (25). If there no unassigned states exist, end the computation.

$$C_{mn} - P_n = \max_{j=1,\cdots,L(k)} \left(c_{mj} - P_j \right) \quad (25)$$

Step4: Release the state previously assigned to y_k^n and assign track y_k^n to the state x_{k+1}^{m}.

Step5: Update the price of the track y_k^n to be

$$P_n = P_n + d_n + \varepsilon \quad (26)$$

where d_n is equal to the difference between the best and second best assignment cost values for the state x_{k+1}^{m}, and ε could be designed to be a constant or a function to prevent a never-ending cycle.

Step6: Return to Step2.

5 Experimental results

We demonstrate the proposed tracker on the dataset of CAVIAR[13] (Context Aware Vision using Image-based Active Recognition). Video “Meet_Split_3rdGuy”has 922 frames containing persons appearing, merging, splitting, or disappearing in a dense scene .

Tab. 1: Parameter values used in experiments

σ_v :standard deviation of state noise	4
σ_w :standard deviation of measurement noise	2
P_D : detection probability	0.96
P_S : survival probability	0.9
λ_c : average number of clutter points per frame	0.01
u : probability distribution of each clutter	$(380*280)^{-1}$
U : merging threshold of Gaussian terms	5
T : truncation threshold of Gaussian terms	0.2
Δ : sampling time	1s

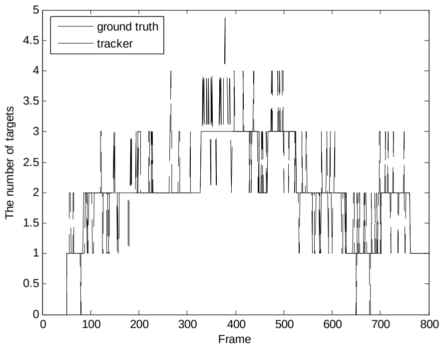


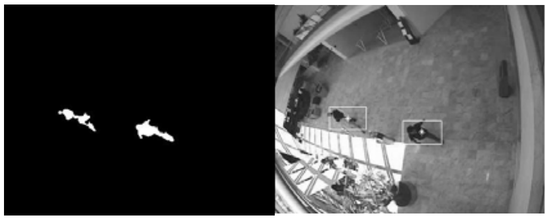
Fig.2: Comparison number estimates with the ground truth. The dashed line is the number estimates by PHD filter. The solid line is the ground truth number of targets.

The important parameters used in the experiment are presented in Table 1. Fig.2 illustrates the comparison result between the correct number estimates for the first 800 frames and ground truth. The accuracy for number estimate is 82.75%. The errors maybe come from the inaccuracy of detections like clutter and false measurements of position and scales, or from that PHD filter usually has a slower response in confirming new targets due to the trade off between clutter removal and response time.

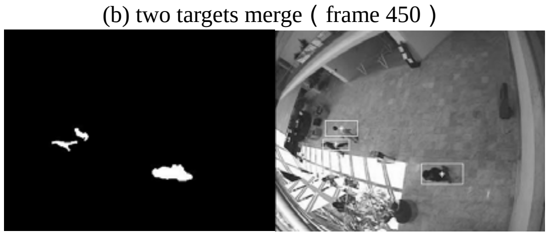
The left images of Fig.3 show the results of detection, while the right 4 frames show the results of PHD filter on the detections with white stars indicating position estimates of centriods and white rectangle indicating scale estimates. When detections are used to update, Gaussian components with associated weights for PHD start growing around them. Multiple coherent and consecutive detections are necessary to increase the weights to a level greater than state extraction threshold. Therefore, the PHD filter will abandon the Gaussian components around false detections (clutter) when the clutter is not persistent, but reserve new Gaussian components around targets generating detections. As shown by the tracking results of birth, merging, splitting and disappearing targets, the proposed algorithm manage to track a variable number of targets.

Fig.4 provides the tracking results for frames from 251 to 501 of Video "Meet_Split_3rdGuy". Tracks of the 3 targets are plotted by there different marker specifiers, namely white point, white square and black point respectively. Comparison of results between Fig.4 (a) (c) and Fig.4 (b) (d) indicates that the proposed multi-target tracking framework can clear clutter and wrong tracks effectively. Fig.4 (b) and Fig.4 (d) show that combined with the proposed association method, PHD filter is able to track a target after a total occlusion and recognize its trajectory.

(a) one birth target（frame 330）



(b) two targets merge（frame 450）



(c)two targets split（frame 468）

(d)one target disappearing（frame528）

Fig.3: Detection and PHD filtering results

In terms of VACE protocol[14] and the available ground-truth data of sequence "Meet_Split_3rdGuy", scores of Multiple Object Tracking Accuracy (MOTA) and Tracking Precision (MOTP) are computed for tracker with PHD filter (P-Tracker) and without PHD filter (Tracker). The comparison result given in Tab.2 indicates that P-Tracker outperforms Tracker.

Tab. 2: Comparison result of tracker with PHD filter (P-Tracker) and without PHD filter (Tracker)

Measures	P-Tracker	Tracker
MOTP	0.68	0.61

MOTA	0.58	0.53

In our experiments, the proposed multi-target tracking algorithm was carried out using an Intel 2.2GHz CPU PC. Detection was done at a rate of 9.6 frames per second for 384*288 images while filtering and recognition were finished at a rate of 16.4 frames per second without code optimization. The computational complexity for GM-PHD filter at time $k+1$ is given by $(J_k(1+J_{\beta,k+1})+J_{\gamma,k+1})(1+|Z_{k+1}|)=o(J_k|Z_{k+1}|)$, i.e., the computational load is linearly proportional to the number of Gaussian components J_k at time k, the number of Gaussian components $J_{\beta,k+1}$ for spawned targets and $J_{\gamma,k+1}$ for birth targets at time $k+1$, and measurement number $|Z_{k+1}|$ at time $k+1$.

(a) frame 371(Tracker) (b) frame 371(P-Tracker)

(c) frame 501(Tracker) (d) frame 501(P-Tracker)

Fig.4: Tracking results on detections using tracker with PHD filter (P-Tracker) and without PHD filter (Tracker).

6 Conclusions

In this paper, we combine the proposed association method with the probability hypothesis density filter (PHD) to build a multi-target visual tracking framework. Foreground objects are detected and analyzed by the motion detector to generate observations, and states with position and scale for a varying number of targets are estimated using GM-PHD filter. Then the state estimates are input to the proposed association method based on Auction algorithm and output tracks with identity. Experimental results over a real-world sequence demonstrate that the proposed method can track a variable number of targets and identify their trajectory in video efficiently.

References

[1] Bar-Shalom Y, Fortmann T. *Tracking and Data Association*[M]. San Diego: Academic Press, 1988:4-10.D. Z. Cheng. Matrix and Polynomial Approach to Dynamic Control Systems[M]. Beijing: Science Press, 2002.

[2] Bar-Shalom Y, Li X R, and Kirubarajan T. *Estimation with applications to tracking and navigation*[M]. New York: Wiley, 2001:199-261.

[3] Arulampalam M S, Maskell S, Gordon N, and Clapp T. *A tutorial on particle filters for online non-linear/non-Gaussian Bayesian tracking*[J]. IEEE Transactions on Signal Processing, 2002, 50 (1):174-188.

[4] Reid D B. *An algorithm for tracking multiple targets*[J]. IEEE Transactions on Automatic Control, 1979, 24(12): 843-854

[5] Singh V K, Wu B, Nevatia R. *Pedestrian tracking by associating tracklets using detection residuals*[A]. In: IEEE Workshop on Motion and video Computing[C], Copper Mountain, Colombia, 2008:8-9.

[6] Nguyen Phi-Vu, Le Hoai-Bac. *Incorporating statistical background model and joint probabilistic data association filter into motorcycle tracking*[A]. In: IEEE International Conference on Research, Innovation and Vision for the Future[C], Ho Chi Minh, Vietnam,2008:284-291.

[7] Mahler R. *Multi-target Bayes filtering via first-order multi-target moments*[J]. IEEE Transactions on Aerospace and Electronic Systems, 2003, 39(4): 1152-1178.

[8] Vo B, Singh S, and Doucet A. *Sequential Monte Carlo implementation of the PHD filter for multi-target tracking*[A]. In: Proceedings of the sixth International Conference of Information Fusion[C], Cairns, Australia, 2003: 792-799.

[9] Vo Ba-Ngu, Ma Wing-Kin. *The Gaussian Mixture Probability Hypothesis Density Filter*[J]. IEEE Transactions On Signal Processing, 2006, 54(11):4091-4104.

[10]Ya-Dong Wang, Jian-Kang Wu, Ashraf A. Kassim, et al. *Data-driven probability hypothesis density filter for visual tracking*[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2008, 18(8): 1085-1095.

[11]Vo B N, Pasha A and Tuan H D. *A Gaussian mixture PHD filter for nonlinear jump Markov models* [A]. In: Proceedings of IEEE Conference on Decision and Control [C], San Diego, CA, USA, 2006: 3162-3167.

[12]D. P. Bertsekas. *The auction algorithm: A distributed relaxation method for the assignment problem*[J]. Annals of operations research, 1988, 14(1): 105-123.

[13]Dataset of European Commission Funded Context Aware Vision Using Image-Based Active Recognition (CAVIAR)Project[DB/OL].http://groups.inf.ed.ac.uk/vision/caviar/caviardata1/. 2007-10-2

[14] R. Kasturi, *Performance evaluation protocol for face, person and vehicle detection & tracking in video analysis and content extraction*, Computer Science and Engineering, University of South Florida, Jan.2006, http://isl.ira.uka.de/clear06/downloads/ClearEval_Protocol_v5.pdf