Multi-Object Tracking through Occlusions by Local Tracklets Filtering and Global Tracklets Association with Detection Responses

Junliang Xing, Haizhou Ai
Computer Science and Technology Dept.
Tsinghua University, Beijing, China
ahz@mail.tsinghua.edu.cn

Shihong Lao
Core Technology Center
Omron Corporation, Kyoto, Japan
lao@ari.ncl.omron.co.jp

Abstract

This paper presents an online detection-based two-stage multi-object tracking method in dense visual surveillances scenarios with a single camera. In the local stage, a particle filter with observer selection that could deal with partial object occlusion is used to generate a set of reliable tracklets. In the global stage, the detection responses are collected from a temporal sliding window to deal with ambiguity caused by full object occlusion to generate a set of potential tracklets. The reliable tracklets generated in the local stage and the potential tracklets generated within the temporal sliding window are associated by Hungarian algorithm on a modified pairwise tracklets association cost matrix to get the global optimal association. This method is applied to the pedestrian class and evaluated on two challenging datasets. The experimental results prove the effectiveness of our method.

1. Introduction

Multi-object tracking is important for many computer vision applications. This is a relatively easy task when objects are isolated and easily distinguished from each other and the background. However, in complex and dense scenarios, many objects may have similar appearance, and occlusions happen very frequently, such as object self-occlusion, occlusion between multiple objects, and occlusion by other scene objects. In these situations, robust multi-object tracking would be a very difficult problem and many traditional object tracking methods may fail. We propose a method that can robustly track multiple objects under such challenging conditions.

Recently with the significant progresses achieved in object detection researches, detection-based tracking methods gain more and more attentions[11][15] since they are essentially more flexible and robust in complex environments no regard of whether camera moves or not and they are fully automatic which do not need outside initialization in tracking which is of great importance in practical applications. However, the accuracy of state-of-the-art object detectors is still far from perfect. The detection performance is usually a tradeoff between the detection rate and the false alarm rate. Missed detections and false alarms and inaccurate responses happen frequently in the detection procedure which provides misleading information to tracking algorithms. Detection-based tracking method must overcome these failures of the detector, and the difficulties caused by occlusions and similar appearance among multiple objects.

The aim of this work is to overcome the limitations of current object detectors to track multiple objects through occlusions online in dense surveillance scenarios. We propose an online detection-based two-stage multi-object tracking method. In the local stage, a particle filter with observer selection that could deal with partial object occlusion is used to generate a set of reliable tracklets. In the global stage, the detection responses are collected from a temporal sliding window to deal with ambiguity caused by full object occlusion to generate a set of potential tracklets. The reliable tracklets generated in the local stage and the potential tracklets generated within the temporal sliding window are associated by Hungarian algorithm on a modified pairwise tracklets association cost matrix to get the global optimal association. The two stages incorporate with each other which make our algorithm seek both the local optimum trajectory for each object and the global optimum trajectories for all the tracked objects.

The main contributions of this paper include: 1) a two-stage online tracking framework which seeks both the local optimum for each object and global optimum for all the tracked objects; 2) a human partition method concentrates on the upper human body in three different levels which is more suitable for human tracking in common visual surveillance scenario; 3) a particle filter with observer selection that could deal with partial object occlusion for generating reliable tracklets; 4) a modified pairwise tracklets association cost matrix for the Hungarian algorithm to solve data association problem which could model tracklets association, initialization, termination and false alarms.

978-1-4244-3991-1/09/$25.00 ©2009 IEEE

The rest of the paper is structured as follows. Related work is discussed in Section 2. The outline of our approach is described in Section 3. Problem formulation is given in Section 4. A detailed description of our approach is given in Section 5. Experimental results are presented in Section 6, and Section 7 concludes the paper.

2. Related work

Detection based tracking algorithms obtain object hypotheses by applying an object detector to images. The detector is learned off-line from labeled training data. Given detection responses generated by the detector, the tracking method needs to retrieve the real objects among those responses and set ID for each of them in every frame. To deal with this data association problem, there are usually two strategies: one is to associate the responses locally (frame by frame) while the other associates them globally.

To associate the responses locally, Wu *et al*. [15] defined an affinity measure between detection responses based on cues from position, size, and color and used a greedy algorithm to associate object hypotheses and detection responses. New object trajectories were initialized whenever the detection responses did not match with any existing trajectories for a certain number of frames; old trajectories were terminated when they were lost by the detector for a certain number of frames. This direct linking method depends a lot on the detector and is sensitive to temporal observation missing. Li *et al*. [10] and Okuma *et al*. [12] used particle filter methods to associate the detection responses of an unknown number of objects. The detection responses were used to generate new particles and evaluate existing particles. By particle filtering, the tracker could maintain multiple hypotheses. However, increasing the number of particles requires more computational cost. To improve the computational efficiency, Li *et al*. [11] used multiple detectors (observers) to form a cascade particle filter. The order in which the detectors were applied was determined based on their computational costs: the faster the earlier. The local association methods used only the information in two consecutive frames which makes them incline to drift when multiple objects are close to each other.

To overcome the drifting problem of local association method, one approach is to optimize multiple trajectories simultaneously such as in Multi-Hypothesis Tracking (MHT) [13] or in Joint Probabilistic Data Association Filters (JPDAF)[3]. Recently some approaches which try to solve this problem globally have been developed, among which Leibe *et al*. [9] used Quadratic Boolean Programming to couple the detection and estimation of trajectory hypotheses, Andriluka *et al*. [2] used Viterbi algorithm to get optimal object sequences, Zhang *et al*. [16] used a cost-flow network to model the MAP data association problem, and Huang *et al*. [6] did data association of detection responses

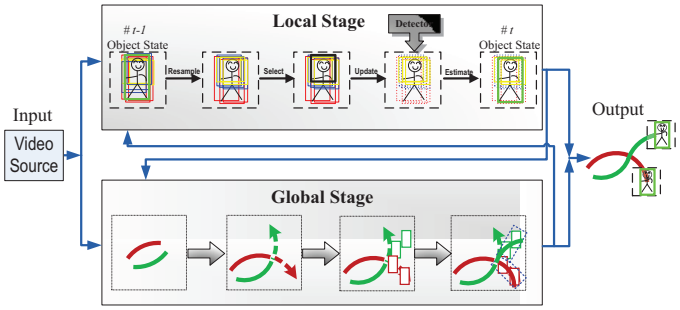

Figure 1. Block diagram of our approach.

hierarchically. As the hypothese search space grows expontially, the global association methods usually need lots of computation which makes them unsuitable for real-time processing.

To our best knowledge, there is not yet work to combine local filtering method and global association method for a better tracking performance in detection based tracking literature. Our work contributes in this direction.

3. Outline of Our Approach

As illustrated in Figure 1, the main idea is a two-stage framework that uses a particle filter algorithm to generate reliable tracklets and uses data association in a temporal sliding window for potential tracklets to optimize object trajectories globally.

In order to deal with partial occlusion problem, an observer selection process is introduced into the particle filter to choose a subset of all observations by the multi-view and multi-part detector which corresponds to all visible object parts.

Tracklets from the particle filter are associated with potential tracklets from a temporal sliding window around trajectory break points of detection responses by Hungarian algorithm based on a modified pairwise tracklets association cost matrix for global trajectory optimization.

4. Problem Formulation

Generally object tracking can be formalized as a sequential Bayesian estimation problem. The trajectories of objects in video are modeled as a sequence of states where each of them basically denotes its location and size. Let $\mathbf{s}_t^i = (p_t^i, s_t^i)$ be the state of a particular object i at frame t where p_t^i is the position and s_t^i is the size, and $\mathbf{s}_{1:t}^i = \{\mathbf{s}_1^i, \ldots, \mathbf{s}_t^i\}$ be the trajectory of the object i up to frame t. Denote $\mathbb{S}_t = \{\mathbf{s}_t^1, \ldots, \mathbf{s}_t^m\}$ as all the object states appeared at frame t and $\mathbb{S}_{1:t} = \{\mathbf{s}_{1:t}^1, \ldots, \mathbf{s}_{1:t}^m\}$ as all object trajectories up to frame t.

Observations are generated by applying a detector on each video frame. Similarly, let $\mathbf{o}_t^i$ be the observation of

the i-th object collected at frame t given the object state $\mathbf{s}_t^i$, and $\mathbf{o}_{1:t}^i = \{\mathbf{o}_1^i, \ldots, \mathbf{o}_t^i\}$ be all the observations of object i up to frame t. Denote $\mathbb{O}_t = \{\mathbf{o}_t^1, \ldots, \mathbf{o}_t^m\}$ as all the observations collected at frame t and $\mathbb{O}_{1:t} = \{\mathbf{o}_{1:t}^1, \ldots, \mathbf{o}_{1:t}^m\}$ as all the observations up to frame t.

For a multi-object tracking system, the final aim is to find the optimal trajectories for all the objects based on the observation set. It is equivalent to maximize the posteriori probability of $\mathbb{S}_{1:t}$ giving the observations $\mathbb{O}_{1:t}$:

$$\mathbb{S}_{1:t}^* = \arg\max_{\mathbb{S}_{1:t}} p(\mathbb{S}_{1:t}|\mathbb{O}_{1:t}) \tag{1}$$

Since the number of all possible enumerations of $\mathbb{S}_{1:t}$ given $\mathbb{O}_{1:t}$ is huge, it prevents a brute force search to find the global optimum. And in fact, global optimization is too time-consuming for practical surveillance applications in which a long time delay is unaffordable. Therefore we turn to a progressive and recuperative strategy. Firstly, we do optimization locally on each object which is equivalent to maximize the posteriori probability of $\mathbf{s}_t^i$ giving the observations $\mathbf{o}_{1:t}^i$:

$$\mathbf{s}_t^{*i} = \arg\max_{\mathbf{s}_t^i} p(\mathbf{s}_t^i|\mathbf{o}_{1:t}^i) \tag{2}$$

This could be done by a particle filter. Denote the tracklet set generated as $\mathbb{S}_{1:t}^+$. Secondly, around trajectory break points of potential tracklets detection responses from a temporal sliding window are generated and associated by greedy linking method [15]. Denote this tracklet set generated as $\mathbb{S}_{1:t}^-$. Further, data association will be done on$\{\mathbb{S}_{1:t}^+, \mathbb{S}_{1:t}^-\}$. This above procedure is summarized as:

$$\begin{aligned}
\mathbb{S}_{1:t}^* &= \arg\max_{\mathbb{S}_{1:t}} p(\mathbb{S}_{1:t}|\mathbb{O}_{1:t}) \\
&= \arg\max_{\mathbb{S}_{1:t}} p(\mathbb{S}_{1:t}|\mathbb{S}_{1:t}^+, \mathbb{S}_{1:t}^-) p(\mathbb{S}_{1:t}^+, \mathbb{S}_{1:t}^-|\mathbb{O}_{1:t}) \\
&= \arg\max_{\mathbb{S}_{1:t}} p(\mathbb{S}_{1:t}|\mathbb{S}_{1:t}^+, \mathbb{S}_{1:t}^-) p(\mathbb{S}_{1:t}^-|\mathbb{S}_{1:t}^+, \mathbb{O}_{1:t}) p(\mathbb{S}_{1:t}^+|\mathbb{O}_{1:t})
\end{aligned} \tag{3}$$

5. Our approach

In common visual surveillance systems where a camera looks down to the plane on a square or a passage where people pass by, occlusion of different types often happens frequently. Occlusion can be classified into three types according to the causation: object self-occlusion, inter-objects occlusion and object occlusion by other scene objects. Object self-occlusion happens when an object changes its pose which may make the detector fail. Inter-objects occlusion, which happens most frequently in dense environment, means that one object is occluded by other objects. Occlusion can also be classified into partial occlusion and full occlusion. If an object is partially occluded, we can still try to get observations to track it. But if an object is fully occluded, no observation would be available. In this paper, we use a multi-view multi-part human detector to collect observations and a particle filter with observer selection process to deal with partial object occlusion. For full object occlusion, we tackle it by data association of the detection responses within a temporal sliding window.

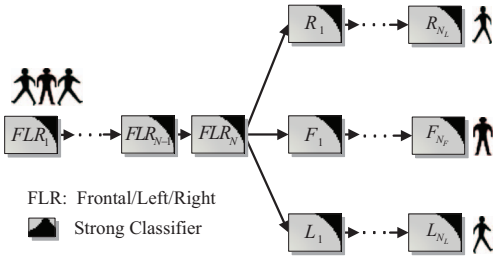

Figure 2. The tree-structure multi-view human detector.

5.1. Multi-View and Multi-Part Human Detector for Observation Collection

For full body human detection we use the algorithm proposed in [5] to train a tree-structure multi-view human detector as illustrated in Figure 2. To deal with partial occlusion of human body, part detectors are also trained to detect the partially occluded humans in which we partition the human body into three levels (head-shoulder (HS), head-torso (HT), full-body (FB)) (as shown in Figure 3) that is concentrated on the upper human body which is different from in [15] where head-shoulder, torso, leg, full-body were used. This partition is preferred considering the following three factors: firstly, in common surveillance scenarios, the upper human body is most likely to be visible when occlusion happens; secondly, the upper human body undergoes less variations compared with other human body parts (such as legs) which makes the upper body centered part detectors learned more robustly; last because of this special partition the three detectors can be efficiently co-trained by feature sharing that guarantees a high computation efficiency in tracking. This partition method describes a human body in three levels of different representation power and trackability. The full-body part covering all the human body area has the highest representation power but the lowest trackability because it's most likely to be occluded by other objects which makes it harder to be observed and tracked especially in crowded situation.

In our system, we use this multi-view multi-part human detector to collect observations. A human hypothesis consists of three overlapped part areas of the human body: the Full Body, the Head-Shoulder, and the Head-Torso. Denote the detection response as a 6-tuple $\mathbf{rp} = \{l, p, s, t, v, a\}$ where l is the label indicating the response type which could be FB, HS or HT, p is the position, s is the size, t is the time stamp (frame index), v is the visible score and a is the appearance model. A combined response is the union of the representations of its parts, $\mathbf{rc} = \{\mathbf{rp}_i | l_i = FB, HS, HT\}$. A human hypothesis H can be represented as $\{\mathbf{rc}, u\}$ where

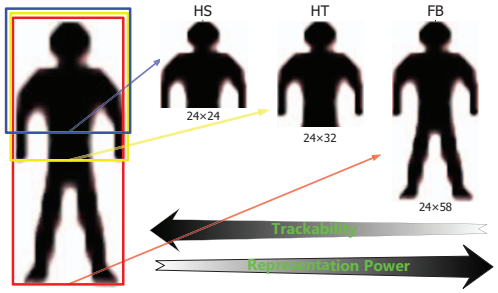

Figure 3. Hierarchy of upper body centered human partition.

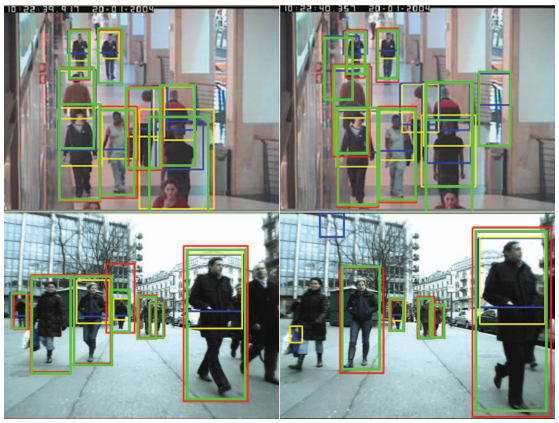

Figure 4. Detection Responses generated by the detector (red for FB, yellow for HT, blue for HS, green is the human hypothesis).

u indicates the visible part. If u is none of the three response types, it means the human is not visible.

In tracking, the detector is applied to each frame to get a set of detection responses. Under the assumption that humans are on a plane to which the camera looks down, the method in [15] is used to check the visibility of each detection response and to propose human hypotheses H. The human hypotheses proposed in each frame are used both in the local tracklets filter stage and the global tracklets association stage later. Figure 4 shows the responses generated by the detector on some sample frames.

5.2. Local Tracklets Filtering

In the local stage, a set of reliable tracklets $\mathbb{S}^+_{1:t}$ ($P(\mathbb{S}^+_{1:t}|\mathbb{O}_{1:t})$) is to be generated. As formalized in section 4, this is equivalent to sequentially estimate $P(\mathbf{s}^i_t|\mathbf{o}^i_{1:t})$, which stands for the distribution of the i-th object state given all the observations $\mathbf{o}^i_{1:t}$(hereafter, we suppress the subscript t and superscript i for briefness when there is no ambiguity).

This can be done by a particle filter or CONDENSATION [7] based on the following well-known two-step recursion procedure:

$$\begin{aligned}\text{Predict}: p(\mathbf{s}_t|\mathbf{o}_{1:t-1}) &= \int p(\mathbf{s}_t|\mathbf{s}_{t-1})p(\mathbf{s}_{t-1}|\mathbf{o}_{1:t-1})d\mathbf{s}_{t-1} \\ \text{Update}: p(\mathbf{s}_t|\mathbf{o}_{1:t}) &\propto p(\mathbf{o}_t|\mathbf{s}_t)p(\mathbf{s}_t|\mathbf{o}_{1:t-1}) \end{aligned} \quad (4)$$

The recursion requires a motion model $p(\mathbf{s}_t|\mathbf{s}_{t-1})$and an observation model $p(\mathbf{o}_t|\mathbf{s}_t)$. To handle complicated distributions which lead to analytical intractability, particle filter approximates the two steps by a set of weighted samples $\{\mathbf{s}^{(n)}_t, \pi^{(n)}_t\}^N_{n=1}$.

When the system has a reliable observation model, the state of the object can be well updated. This corresponds to the situation when one object is isolated from other objects and the detector can generate good observations. But in the dense environment where multiple objects are close to each other and not all of the human bodies are fully visible, the detector will fail to generate reliable observations or even give wrong responses. In these situations, the prediction and update process of particle filter will give unpredictable filtering results.

In order to deal with the unreliable observation problems, we propose to select the best subset of corresponding observations in the particle filter procedure.

Suppose we have l different observations for each state s that is denoted as $\mathbf{o} = (o_1, \ldots, o_l)$, where the k-th observation is $p(o_k|\mathbf{s})$. Assuming observations are conditionally independent given the target state $\mathbf{s}$, we have

$$p(\mathbf{o}|\mathbf{s}) = p(o_1, \ldots, o_l|\mathbf{s}) = \prod_{i=1}^{l} p(o_i|\mathbf{s}) \quad (5)$$

Instead of updating particle weights directly according to the above observation model, we dynamically select the best subset of current observations which corresponds to visible parts. For this purpose, the two-step recursion is extended to a three-step one as follows:

$$\begin{aligned}\text{Predict}: p(\mathbf{s}_t|\mathbf{o}_{1:t-1}) &= \int p(\mathbf{s}_t|\mathbf{s}_{t-1})p(\mathbf{s}_{t-1}|\mathbf{o}_{1:t-1})d\mathbf{s}_{t-1} \\ \text{Select}: \hat{\mathbf{o}}_t &= \arg\max_{\hat{\mathbf{o}}_t \subseteq \mathbf{o}_t} (p(\hat{\mathbf{o}}_t|\mathbf{s}_t)) \\ \text{Update}: p(\mathbf{s}_t|\mathbf{o}_{1:t}) &\propto p(\hat{\mathbf{o}}_t|\mathbf{s}_t)p(\mathbf{s}_t|\mathbf{o}_{1:t-1}) \end{aligned} \quad (6)$$

The Select procedure selects the best subset $\hat{\mathbf{o}}_t = (o_{l^1_t}, \ldots, o_{l^{c_t}_t})$ of all the observations $\mathbf{o}_t$ which maximizes the observation probability based on the predicted object state $\mathbf{s}_t$.

The initialization and termination of a reliable tracklet are both guided by the global stage (which will be described in subsection 5.3). Our particle filter with observer selection process automatically selects the best subset of weighted particle samples according to the object state in previous frame to update the object state in current frame to grow the tracklet. If the object state can not be observed (all the parts are invisible or lost by the observation model), the local tracklet filtering stage on this object stops and the tracklet is buffered for the global stage.

In implementation, a zero order motion model with Gaussian diffusion was chosen for the dynamic model and the confidence output by the detector was used as the observation model [11].

5.3. Global Tracklets Association with Detection Responses

In the global stage, a set of potential tracklets $\mathbb{S}^{-}_{1:t}$ ($P(\mathbb{S}^{-}_{1:t}|\mathbb{S}^{+}_{1:t}, \mathbb{O}_{1:t})$) is to be generated from a temporal sliding window around trajectory break points using the detection responses and then they will be associated with the tracklet set $\mathbb{S}^{+}_{1:t}$ from the particle filter with observer selection to optimize the final object trajectories $\mathbb{S}^{*}_{1:t}$ (see equation 3 in section 4).

5.3.1 Detection Response Association for Potential Tracklets

Detection responses are associated within a temporal sliding window around trajectory break points (tails of reliable tracklets) to generate a set of potential tracklets. The size of the temporal sliding windows can be adjusted according to the occlusion degree of the video data.

At each frame within the temporal sliding window, we first collect object hypotheses according to detection responses. And then we add occluded object hypotheses according to the tracking results at the local stage. These object hypotheses are then associated with the object hypotheses in the previous frame based on the affinity of the responses in position, size and appearance. The association is guided by a greedy strategy as in Wu *et al*. [15]. If an object hypothesis is associated in T (typically $T = 5$) consecutive frames, then a new potential tracklet is generated. All the potential tracklets generated in this way will be further associated with the reliable tracklets as described in the following subsection.

5.3.2 Pairwise Tracklet Association for Final Object Trajectories

The tracklet association approaches [14][8][6] model the joint association likelihood $P(T_1, T_2, \ldots, T_n)$ of tracklets as the product of pairwise association likelihoods $P(T_i, T_j)$. The pairwise association likelihoods are then represented in an association cost matrix $\mathbf{C}$ (with $C_{ij} = -\log(P(T_i, T_j))$ and the optimal association (maximizing the joint likelihood) is computed using the Hungarian algorithm. In our implementation, we made some change mainly to the association cost matrix to meet the requirement of associating two kinds of tracklets as illustrated below.

Supposing there are m tracklets in the Reliable Tracklets set $\mathbb{S}^{+}_{1:t}$ and n tracklets in the Potential Tracklets set $\mathbb{S}^{-}_{1:t}$. There may be four situations after tracklet association between these two independent tracklet sets: TR_i associates with TP_j; TR_i is a terminal object trajectory; TP_j is an initial object trajectory; TP_j is a false trajectory. In order to model all these four situations, we define tracklet association cost matrix as follows:

$$\mathbf{C} = \begin{bmatrix} A_{m\times n} & B_{m\times m} \\ D_{n\times n} & \mathbf{0}_{m\times n} \end{bmatrix} \tag{7}$$

where $A = \{a_{ij}\}_{m\times n}$ models the first situation, $a_{ij} = -\log P(TR_i, TP_j)$ is the pairwise association cost of tracklet TR_i and TP_j; $B = diag\{b_1, \ldots, b_m\}$ models the second situation, $b_i = -\log P(TR_i)$ is the cost of terminating the Reliable Tracklets TR_i. $D = diag\{d_1, \ldots, d_n\}$ models the third and fourth situation, $d_i = -\frac{1}{2}(\log P_{init}(TP_j) + \log P_{false}(TP_j))$ is the cost of initializing the Potential Tracklet TP_j as a new track or a false track.

In order to calculate the association likelihood between two tracklets, we explore the appearance, shape and motion attributes of the tracklet. So we describe a tracklet T_i by a 3-tuple $\{\mathbf{A}_i, \mathbf{S}_i, \mathbf{M}_i\}$, where $\mathbf{A}_i$ is the appearance model, $\mathbf{S}_i$ is the shape model and $\mathbf{M}_i$ is the motion model.

Appearance Model $\mathbf{A}_i$: We use color histograms as the appearance model of a tracklet. The color histograms are calculated for each object part area and are updated over the detection responses along the tracklet when the corresponding part is visible. A weighted Bhattacharya distance using the gaussian kernel between the two tracklets color models is calculated as the distance measure $d(a_i, a_j)$ between the tracklets.

$$P_a(T_i, T_j) = \exp(-d(a_i, a_j)) \tag{8}$$

where $d(a_i, a_j) = mean\{d(a_i^L, a_j^L)|L = FB, HT, HS\}$

Size Model $\mathbf{S}_i$: Because the aspect ratio of an object is a constant (determined by the aspect ratio of the training samples), we just compute the estimated 3D object height of each tracklet from the responses along the tracklet as the object shape model. The expected 3D height averaged over the entire tracklet is used as the 3D height estimation.

$$P_s(T_i, T_j) = \exp(-\frac{|h_i - h_j|}{h_i + h_j}) \tag{9}$$

Motion Model $\mathbf{M}_i$: We calculate both the forward velocity and backward velocity of the tracklet as its Motion Model. The forward velocity is calculated from the refined position of the tail response of the tracklet while the backward velocity is calculated from the refined position of the head response of the tracklet. Motion model in forward direction is represented by a gaussian $\{x_i^F, \Sigma_i^F\}$, and in backward direction by a gaussian $\{x_i^B, \Sigma_i^B\}$.

$$\begin{aligned} P_m(T_i, T_j) = & G(p_i^{tail} + v_i^F \Delta t; p_j^{head}, \Sigma_j^B) \cdot \\ & G(p_j^{head} + v_j^B \Delta t; p_i^{tail}, \Sigma_i^F) \end{aligned} \tag{10}$$

where p_i^{head} is the position of the head response of tracklet T_i and p_i^{tail} is the position of the tail response of tracklet T_i.

Assuming the independence among the three tracklet models, the pairwise association likelihood of tracklet and can be calculated as:

$$P(TR_i, TP_j) = P_a(TR_i, TP_j) \cdot P_s(TR_i, TP_j) \cdot P_m(TR_i, TP_j) \quad (11)$$

The likelihood of terminating the Reliable Tracklet TR_i is modeled as:

$$P_{term}(TR_i) = Z_t(1-\alpha)^{\omega - b} \quad (12)$$

where Z_t is a normalization factor, ω is the temporal sliding window size, b is the number of frames in which the object is buffered because of occlusion.

The likelihood of the Potential Tracklet being a new track is modeled as:

$$P_{init}(TP_i) = Z_t(1-\alpha)^{l} \quad (13)$$

where l is the length of the Potential Tracklet TP_i, and being a false track as:

$$P_{false}(TP_i) = Z_t(1-\alpha)^{2T-l} \quad (14)$$

So far we can apply the Hungarian algorithm on cost matrix $\mathbf{C}$ to get optimal association. Denoting $M^* = [m_{ij}^*]_{(m+n)(m+n)}$ as the optimal assignment matrix obtained, for each $m_{ij}^* = 1$, do as follows:

(1).If $i \leq m$ and $j \leq n$, then associate TR_i and TP_j;

(2).If $i \leq m$ and $j > m$,then TR_i is considered as a terminated tracklet;

(3).If $i > m$ and $j \leq m$, then TP_j is considered as a new track if $P_{init}(TP_i) > P_{false}(TP_i)$, or a false track if $P_{init}(TP_i) < P_{false}(TP_i)$;

Table 1 summarizes the overall algorithm of the two stage tracking framework.

6. Experiments

We evaluated our two-stage multi-object tracking algorithm on two public datasets: the CAVIAR dataset [1] and the ETH Mobile Scene (ETHMS) dataset [4]. The CAVIAR dataset includes 26 video sequences of a walkway in a shopping center taken by a single camera with frame size of 384×288 and frame rate of 25fps. The ETHMS dataset includes 4 video sequences of street scenes taken by a moving camera, with frame size of 640×480 and frame rate of 15fps. Both of the two datasets are very challenging because of the heavy inter-person occlusions and poor image contrast between objects and background. We evaluated our algorithm on its tracking performance, detection performance and speed, and compared our results with Wu *et al.*'s [15] method and Zhang *et al.*'s [16] work because they have reported the best results on the datasets.

Table 1. Algorithm of the two-stage tracking framework.

Given: the tracked object set $\mathbb{S}_{t-1} = \{\mathbf{s}_{t-1}^1, \ldots, \mathbf{s}_{t-1}^m\}$ with their observations and other attributes at frame $t-1$, the temporal sliding window buffer W.

Local Stage: For each tracked object:

If $\mathbf{s}_{t-1}^i$ can be observed:

- **Resample**: for $s = l_{t-1}^1, \ldots, l_{t-1}^{c_{t-1}}$, simulate $\alpha_{n,o_s} \sim \{\pi_{t-1,o_s}^{i,(n)}\}_{n=1}^{N_{o_s}}$, and repace $\{\mathbf{s}_{t-1,o_s}^{i,(n)}, \pi_{t-1,o_s}^{i,(n)}\}_{n=1}^{N_{o_s}}$ with $\{\mathbf{s}_{t,o_s}^{i,(\alpha_{n,o_s})}, \frac{1}{N_{o_s}}\}_{n=1}^{N_{o_s}}$
- **Predict**: simulate $\mathbf{s}_{n,o_s}^{i,(n)} \sim p(\mathbf{s}_t^i | \mathbf{s}_{t-1,o_s}^{i,(n)})$
- **Select**: $\hat{\mathbf{o}}_t^i = Select(\mathbf{o}_t^i) = [o_{l_t^1}, \ldots, o_{l_t^{c_t}}]$
- **Update**:$\pi_{t-1,o_s}^{i,(n)} \leftarrow p(\hat{\mathbf{o}}_t^i | \mathbf{s}_t^{i,(n)})$
- **Estimate**:$\hat{\mathbf{s}}_t^i = \sum_s \sum_n \mathbf{s}_t^{i,(n)} \cdot \pi_{t,o_s}^{i,(n)}$

Else buffer the object for the global stage.

Global Stage:

- Slide W tail to current frame;
- Retrieve the reliable tracklets from the local stage;
- Retrieve the potential tracklets larger than T within W and then smooth the position and size using Kalman Filter;
- Calculate the modified pairwise association cost matrix and apply Hungarian algorithm;
- Manage the object trajectories according to the assignment matrix;

Output:Estimate the objects state $\hat{\mathbb{S}}_t = \{\hat{\mathbf{s}}_t^1, \ldots, \hat{\mathbf{s}}_t^m\}$

Table 2. Comparison of the tracking results on CAVIAR dataset.

Method	GT	MT	PT	ML	FRMT	IDS
Wu *et al.* [15]	140	106	25	9	35	17
Zhang *et al.* [16] Alg.1	140	104	29	7	58	7
Zhang *et al.* [16] Alg.2	140	120	15	5	20	15
Our Local Stage	140	112	12	6	33	16
Our two-stage	140	118	17	5	24	14

6.1. Tracking Performance

We adopt the same metrics as in [15] to evaluate the tracking performance:

- **MT**: Mostly Tracked trajectories, the number of trajectories that are successfully tracked for more than 80%;
- **ML**: Mostly Lost trajectories, the number of trajectories that are tracked for less than 20%;
- **PT**: Partially Tracked trajectories, the number of trajectories that are tracked between 20% and 80%;
- **FRMT**: Fragmentation, the number of times a trajectory is interrupted;
- **IDS**: ID Switches, the number of times two trajectories switch their IDs.

The tracking results on the CAVIAR dataset are showed in Table 2. As in [16], people that are too small in the images (less than 24 pixels in width) are not counted in the evaluation. This dataset is fully independent from the training dataset of our detector or tracker.

Table 3. Comparison of the detection results.

Dataset	Method	DR	DA	FAF
CAVIAR	Input[15][16]	72.8%	N/A	0.270
	Wu *et al*. [15]	75.2%	N/A	0.281
	Zhang *et al*. [16] Alg.1	75.2%	N/A	0.081
	Zhang *et al*. [16] Alg.2	74.3%	N/A	0.105
	Our Input	76.7%	0.657	0.262
	Our Local Stage	79.3%	0.739	0.078
	Our two-stage	81.8%	0728	0.136
ETHMS	Input[16]	64.3%	N/A	1.51
	Zhang *et al*. [16] Alg.1	68.3%	N/A	0.85
	Zhang *et al*. [16] Alg.2	70.4%	N/A	0.97
	Our Input	65.1%	0.568	1.239
	Our Local Stage	71.8%	0.658	0.762
	Our two-stage	75.2%	0.643	0.939

From Table 2 we can see that the tracking results of our local stage method outperform both the results of Algorithm.1 in [16] and the results in [15] in every aspect which reflect the effectiveness of our local stage. Though we only collect global information within a temporal sliding window in our two-stage tracking framework, we get comparable results to the Algorithm.2 in [16]. Figure 5 shows some tracking results from which we can find that our two-stage method can not only track partial occluded humans but also can recover trajectories from full occlusion.

6.2. Detection Performance

We use the detection rate (DR), detection accuracy (DA) and false alarm per frame (FAF) to evaluate detection performance and compare with direct detection result and previous methods[15][16]. The detection accuracy is defined as:

$$DA = \frac{\sum_i A(D_i \cap G_i)}{\sum_i A(D_i \cup G_i)} \tag{15}$$

where G_i is the i-th ground truth (bounding box) while D_i is the corresponding detection result. $A(\cdot)$ calculates the area of the input region.

The results are showed in Table 3 from which we can see that our input observation set has a much higher DR and relative lower FAR than the input observation set in [15][16]. This proves the adaptability of our human body partition method in common visual surveillance system. Compared to our input observation set, both the local stage and two-stage tracking method make improvement on the detection rate, detection accuracy and false alarm per frame.

6.3. Speed

The speed of our system is about 4fps on the CAVIAR dataset and about 1.5fps on the ETHMS dataset which includes frame by frame detection time. The test machine is an Intel Core 2 CPU with 2G RAM and the program is coded in C++ without any parallel procedure. We analyzed the execution time of our program and found that the detection for collecting observations took up about 80% of the total execution time and the speed of frame by frame detection is less than 5fps on the CAVIAR dataset. To improve the detection efficiency, we make it scan only part of the image in each frame which results in a tracking that could run at 15fps without losing much accuracy. So given an input video of size 384×288, the system can process it online smoothly. The two-stage tracking system has a very high tracking efficiency which could be used for online visual surveillance system.

7. Conclusions

In this paper, we present an online detection-based two-stage multi-object tracking framework which seeks both the local optimum trajectory for each object and the global optimum trajectories for all the tracked objects. It integrates particle filter based tracker with data association based tracker efficiently to guarantee online tracking performance. It is not only different from the classical particle filter which results in many breaks of object trajectories due to occlusions but also different from the conventional association based tracker which is time consuming and usually offline. Experimental results on two challenging datasets show that the proposed method significantly improves the tracking performance both in tracking accuracy and tracking efficiency in dense visual surveillance environment with different types of occlusions.

8. Acknowledgement

This work is supported in part by National Basic Research Program of China (2006CB303102), Beijing Educational Committee Program (YB20081000303), and it is also supported by a grant from Omron Corporation.

References

[1] Caviar dataset. http://homepages.inf.ed.ac.uk/rbf/CAVIAR/. 6

[2] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. In *CVPR*, 2008. 2

[3] I. J. Cox. A review of statistical data association for motion correspondence. *IJCV*, 10(1):53–66, 1993. 2

[4] A. Ess, B. Leibe, and L. V. Gool. Depth and appearance for mobile scene analysis. In *ICCV*, 2007. 6

[5] C. Hou, H. Ai, and S. Lao. Multiview pedestrian detection based on vector boosting. In *ACCV*, 2007. 3

[6] C. Huang, B. Wu, and R. Nevatia. Robust object tracking by hierarchical association of detection responses. In *ECCV*, 2008. 2, 5

[7] M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. *IJCV*, 28(1):5–28, 1998. 4

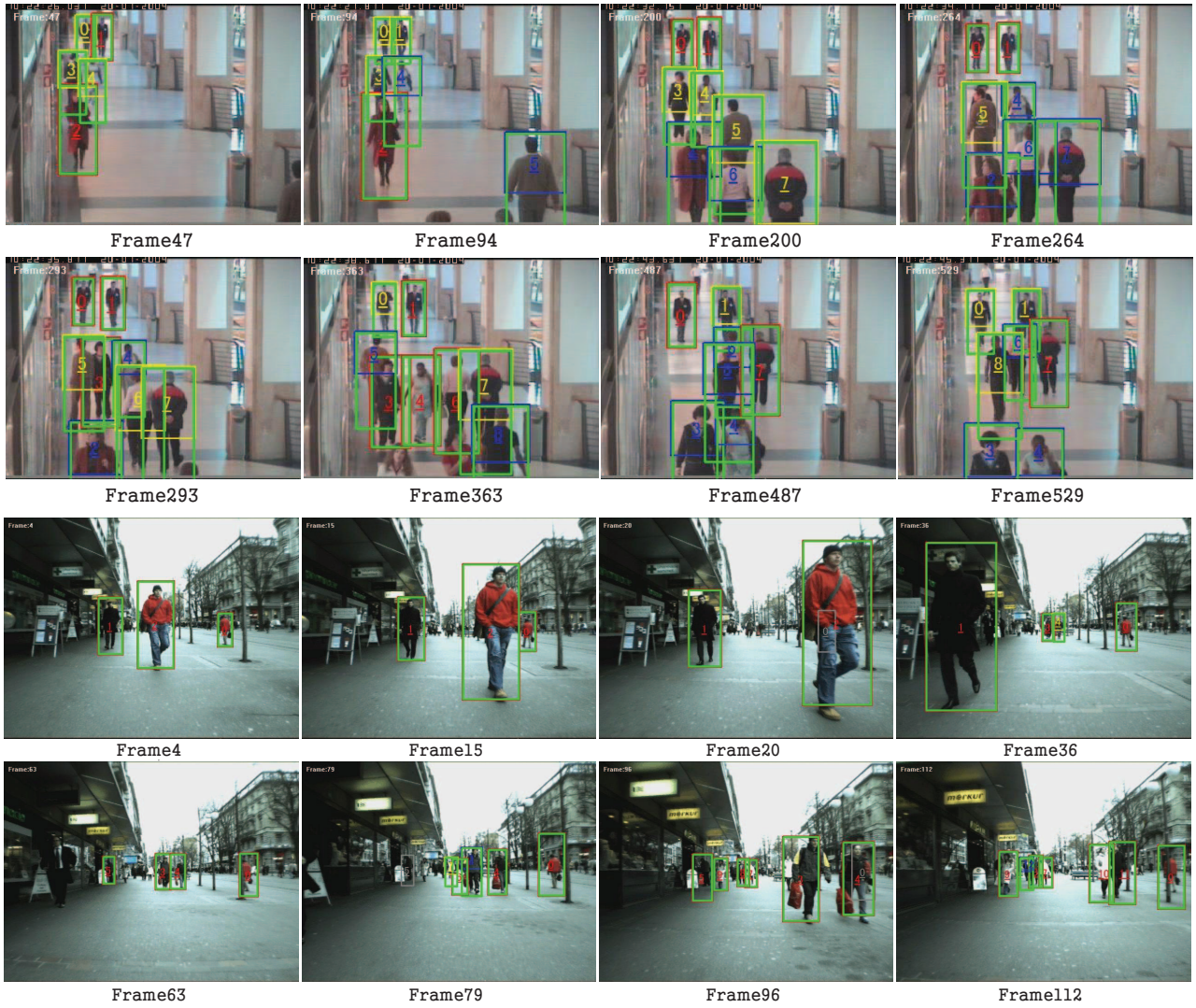

Figure 5. Tracking result, the upper two rows are in CAVIAR, the bottom two rows are in ETHMS. (rectangle in red, yellow or blue indicates the largest part used for tracking, green rectangle is the local tracking result, gray rectangle is the global tracking result).

[8] R. Kaucic, A. G. A. Perera, G. Brooksby, J. Kaufhold, and A. Hoogs. A unified framework for tracking through occlusions and across sensor gaps. In *CVPR*, 2005. 5

[9] B. Leibe, K. Schindler, and L. V. Gool. Coupled detection and trajectory estimation for multi-object tracking. In *ICCV*, 2007. 2

[10] Y. Li, H. AI, C. Huang, and S. Lao. Robust head tracking based on a multi-state particle filter. In *FG*, 2006. 2

[11] Y. Li, H. AI, T.Yamashita, S.Lao, and M. Kawade. Tracking in low frame rate video: A cascade particle filter with discriminative observers of different lifespans. In *CVPR*, 2007. 1, 2, 5

[12] K. Okuma, A. Taleghani, D. Freitas, J. J. Little, and D. G. Lowe. A boosted particle filter: Multitarget detection and tracking. In *ECCV*, 2004. 2

[13] D. Reid. An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, 24(6):843–854, Dec 1979. 2

[14] C. Stauffer. Estimating tracking sources and sinks. In *IEEE Workshop on Event Mining in Video*, 2003. 5

[15] B. Wu and R. Nevatia. Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors. *IJCV*, 75(2):247–266, 2007. 1, 2, 3, 4, 5, 6, 7

[16] L. Zhang, Y. Li, and R. Nevatia. Global data association for multi-object tracking using network flows. In *CVPR*, 2008. 2, 6, 7