

Tracking of Multiple, Partially Occluded Humans based on Static Body Part Detection

Bo Wu and Ram Nevatia
University of Southern California
Institute for Robotics and Intelligent Systems
Los Angeles, CA 90089-0273
{bowu|nevatia}@usc.edu

Abstract

Tracking of humans in videos is important for many applications. A major source of difficulty in performing this task is due to inter-human or scene occlusion. We present an approach based on representing humans as an assembly of four body parts and detection of the body parts in single frames which makes the method insensitive to camera motions. The responses of the body part detectors and a combined human detector provide the “observations” used for tracking. Trajectory initialization and termination are both fully automatic and rely on the confidences computed from the detection responses. An object is tracked by data association if its corresponding detection response can be found; otherwise it is tracked by a meanshift style tracker. Our method can track humans with both inter-object and scene occlusions. The system is evaluated on three sets of videos and compared with previous method.

1 Introduction

Tracking humans in videos is important for many applications, such as visual surveillance, and human computer interaction. There are many sources of difficulty in performing this task. First, the objects to be tracked need to be detected; this is not difficult for moving, isolated humans viewed with a fixed camera and fixed or slowly varying illumination. However, in presence of multiple humans with inter-object occlusion and/or a moving camera, detecting humans reliably becomes a difficult problem. Then, we need to track the humans across the different frames with varying amounts of inter-object or scene occlusions. The image appearance of the objects changes not only with the changing viewpoints but even more strongly with the visible parts of the body and clothing. Also, it is more likely that the identities of objects may be switched during tracking when humans are close to each other.

We describe a method to automatically track multiple, partially occluded humans in a walking or standing pose. We use a part based representation so that occlusions do not affect the entire description as they would in a global representation. Part-based representation has been used for human detection in a single image in some recent work, *e.g.* in [1, 2, 3] but these methods do not use the parts for tracking. In [4], a part-based representation is used for segmenting motion blobs by considering various articulations and their appearances but parts are not tracked explicitly. Part tracking has been used to track the *pose* of a single human, *e.g.* [5, 6], but not *locations* of multiple humans. In our approach, we track the individual detected parts and then combine their responses in a *combined tracker*. The advantage of this approach comes from the observation that under partial occlusion conditions, some parts of the object remain visible and distinguishable and can provide reliable cues for tracking. Of course, when the object is fully occluded, the tracks only be inferred from the observables before and after.

For an automatic multiple object tracking system, three main problems need to be addressed: 1) when to initialize a trajectory? 2) how to track an object? and 3) when to terminate a trajectory? Our approach relies on single frame human detection responses to answer all these three questions. We do not rely on background modeling, hence our method does not require any special preprocessing for moving and/or zooming cameras. The detector is a generalization of our previous work [1] to multi-view conditions. We illustrate the performance of the system on some examples.

1.1 Related work

The literature on human detection from static images and on human tracking from videos is abundant. Many methods for static human detection represent a human as an integral whole, *e.g.* [7, 8, 9, 10, 11, 12]. Some methods for representation as an assembly of parts have also been developed, *e.g.* [1, 2, 3]. Some use global features as in [7, 9, 10]; others

use local features [1, 2, 3, 7, 8, 11, 12]. The integral representation methods with global features, do not work well in presence of occlusion; local feature methods are less sensitive as only some of the features are affected by occlusions. However, only [1] and [7] incorporate explicit inter-object occlusion reasoning.

The method of Leibe *et al.* [7] has two main steps. The first one generates hypotheses by evidence from local features, while the second one verifies the hypotheses by constraints from the global features. These two steps are applied iteratively to compute a local maximum of the image likelihood. In [1] we proposed a human detection method by combining part detection responses. Human body is divided into four parts, full-body, head-shoulder, torso, and legs. For each part a *cascade* detector is learned by boosting *edgelet* features. Then a joint likelihood for multiple humans, which is calculated based on occlusion reasoning, is maximized to find the best interpretation of the detection responses. The detection system in this paper is built on that in [1].

For human tracking, some of the previous methods [4, 13, 14, 15] try to fit multiple object hypotheses to explain the foreground or motion blobs. Because the hypotheses space is usually of high dimension, an efficient optimization algorithm, such as a particle filter [15], MCMC [4, 13] or EM [14] are used. These methods deal with occlusion by computing joint image likelihood of multiple objects. All of these methods have shown experiments with a stationary camera only, where the background subtraction is relatively robust. Some other methods [12, 16] build deformable silhouette models for pedestrians and track the models from edge features. The silhouette matching can be done frame by frame. These methods are less dependent on the camera motion. However they have no explicit occlusion reasoning. None of the above tracking methods deal with occlusion by scene objects explicitly.

1.2 Outline of our approach

Our tracking method is based on tracking parts of the human body. The parts are detected in each frame, treated as a single image, to avoid the necessity of computing reliable motion blobs and to be able to detect static humans as well as the moving ones. We use the four-part representation as in [1]. The responses from the static part detectors are taken as the inputs for the tracker. In [1] only the frontal/rear view is considered. The current system includes left and right profile views. The detection system consists of four part detectors and a combined detector. The performance of the combined detector is better than that of any single part detector in terms of the false alarm rate. However the combined detector does explicit reasoning only for inter-object partial occlusion, while the part detector can work in the presence of both inter-object and scene occlusions.

We track humans by data association, *i.e.* matching the object hypotheses with the detected responses, whenever corresponding detection responses can be found. We match the hypotheses with the combined detection responses first, as they are more reliable than the responses of the individual parts. If this fails, then we try to associate the hypotheses with the part detection responses. If this fails again, a meanshift tracker [18] is used to follow the object. Most of the time, objects are tracked successfully by data association. The meanshift tracker gets utilized occasionally and for short periods only. Since our method is based on the part detection, it can work under both scene and inter-object occlusion conditions. Also, as the cues for tracking are strong in our system, we do not utilize statistical sampling techniques as in some of the previous work, *e.g.* [4, 13, 15]. We initialize a trajectory when evidence from new observations can not be explained by current hypotheses, as also in many previous methods [4, 13, 14, 15, 16]. Similarly, a trajectory is terminated when it is lost by the detectors for a certain period.

The rest of the paper is organized as follows: Section 2 describes our multi-view detection system; Section 3 defines the representation of human and body parts; Section 4 gives the details of our part detection based tracking method; Section 5 shows the experiment results; and Section 6 sums up.

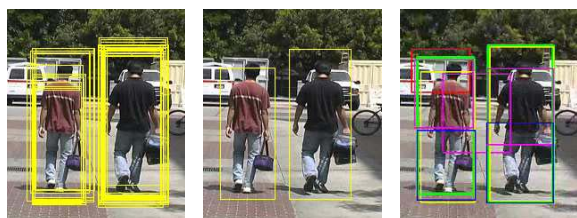
2 Multi-View Human Detection System

The detection system in [1] only considers humans from frontal/rear view point. We extended it to include left and right profile views. For each body part, two detectors are learnt: one for the left profile view, and one for the frontal/rear view (the detector for right profile view is generated by flipping the left profile view one horizontally). Edgelet features are used in a nested cascade detector learned by using boosting methods, as in [1]; a variation of the original framework proposed by Viola and Jones [17]. The training set contains 1,700 positive samples for frontal/rear views, 1,120 for left profile view, and 7,000 negative images. Because our positive training samples are captured from a variety of camera tilt angles, the learned detectors can work with a tilt angle within about $[0^\circ, 45^\circ]$ (0° is the horizontal view point).

For detection, the input image is scanned by all three detectors and the union of their responses is taken as the multi-view detection result. The responses of the detection system have three levels. The first level is a set of the *original responses* of the detectors. In this set, one object may have multiple corresponding responses, see Figure 1(a). The second level is that of the *merged responses*, which are the results of applying a clustering algorithm to the original responses. In this set, one object has at most one corresponding response, see Figure 1(b). The third level is

that of the *combined responses*. The second level results of all parts are the input of a combined detector [1], which computes the joint image likelihood of multiple humans by inter-object occlusion reasoning, and searches the hypotheses space to find the best interpretation. The output of the combined detector is a set of hypotheses, each of which has several matched part responses, see Figure 1(c) for an example. Note, the detection response may not be highly accurate spatially, because the training samples include some background region in order to cover some position and size variations.

Compared to the generative method, *e.g.* [4], the two main limits of this system are the dependence on viewpoints (the tilt angle not exceeding about 45°) and the need of relative high resolution (not smaller than the size of training samples 24×58 pixel).



(a) original responses (b) merged responses (c) combined responses

Figure 1. Static detection responses. a) and b) are from the full-body detector; c) is from the combined detector (green for combined; yellow for full-body; red for head-shoulder; purple for torso; blue for legs).

3 Part and Human Representation

Both the original and the merged responses are called *part responses*. We represent a part response by a 5-tuple, $\mathbf{rp} = \{l, \mathbf{p}, s, f, \mathbf{c}\}$, where l is a label indicating the part type; \mathbf{p} is the image position (x, y) of the part; s is the size; f is a real-valued detection confidence; and \mathbf{c} is an appearance model. The first four elements, l, \mathbf{p}, s , and f , can be obtained from the detection process directly. The appearance model, \mathbf{c} , is implemented as a color histogram; computation and update of \mathbf{c} is described, in detail, later in Section 4.3. The representation of a combined response is the union of the representations of its parts, with an additional *visibility score* v for each part, obtained from occlusion reasoning [1]. If v is smaller than a threshold, θ_v , the part is considered occluded by other objects. Formally $\mathbf{rc} = \{l_i, \mathbf{p}_i, s_i, f_i, \mathbf{c}_i, v_i\}_{i=FB,HS,T,L}$, where the index FB, HS, T, L stands for “full-body”, “head-shoulder”, “torso”, and “legs” respectively.

Objects are detected frame by frame. In order to decide whether two responses, \mathbf{rp}_1 and \mathbf{rp}_2 , from different frames belong to one object, an affinity measure is defined

$$A(\mathbf{rp}_1, \mathbf{rp}_2) = A_{pos}(\mathbf{p}_1, \mathbf{p}_2)A_{size}(s_1, s_2)A_{appr}(\mathbf{c}_1, \mathbf{c}_2) \quad (1)$$

where A_{pos} , A_{size} , and A_{appr} are affinities based on position, size, and appearance respectively. Their definitions are

$$\begin{aligned} A_{pos}(\mathbf{p}_1, \mathbf{p}_2) &= \gamma_{pos} \exp \left[-\frac{(x_1 - x_2)^2}{\sigma_x^2} \right] \exp \left[-\frac{(y_1 - y_2)^2}{\sigma_y^2} \right] \\ A_{size}(s_1, s_2) &= \gamma_{size} \exp \left[-\frac{(s_1 - s_2)^2}{\sigma_s^2} \right] \\ A_{appr}(\mathbf{c}_1, \mathbf{c}_2) &= B(\mathbf{c}_1, \mathbf{c}_2) \end{aligned} \quad (2)$$

where $B(\mathbf{c}_1, \mathbf{c}_2)$ is the Bhattachayya distance between two color histograms and γ_{pos} and γ_{size} are normalizing factors. The affinity between two combined responses, \mathbf{rc}_1 and \mathbf{rc}_2 , is the average of the affinity between their common visible parts

$$A(\mathbf{rc}_1, \mathbf{rc}_2) = \frac{\sum_{i \in PT} A(Pt_i(\mathbf{rc}_1), Pt_i(\mathbf{rc}_2))I(v_{i1}, v_{i2} > \theta_v)}{\sum_{i \in PT} I(v_{i1}, v_{i2} > \theta_v)} \quad (3)$$

where $PT = \{FB, HS, T, L\}$, $Pt_i(\mathbf{rc})$ returns the response of the part i of the combined response \mathbf{rc} , v_{ij} is the visibility score of $Pt_i(\mathbf{rc}_j)$, $j = 1, 2$, and I is an indicator function. The above affinity functions encode the position, size, and appearance information.

4 Human Tracking based on Detection

Before giving our method for trajectory initialization, object tracking, and trajectory termination, we first introduce the data association method which is common to the three modules.

4.1 Data Association

The task of data association is to match the detection responses with the human hypotheses. We use a greedy algorithm to do this. Suppose at time t , we have n hypotheses H_1, \dots, H_n , whose predictions at time $t+1$ are $\widehat{\mathbf{rc}}_{t+1,1}, \dots, \widehat{\mathbf{rc}}_{t+1,n}$, and at time $t+1$ we have m responses $\mathbf{rc}_{t+1,1}, \dots, \mathbf{rc}_{t+1,m}$. First we compute the $m \times n$ affinity matrix \mathbf{A} of all $(\widehat{\mathbf{rc}}_{t+1,i}, \mathbf{rc}_{t+1,j})$ pairs, *i.e.* $\mathbf{A}(i, j) = A(\widehat{\mathbf{rc}}_{t+1,i}, \mathbf{rc}_{t+1,j})$. Then in each step, the pair, denoted by (i^*, j^*) , with the largest affinity is taken as a match and the i^* -th row and the j^* -th column of \mathbf{A} are deleted. This procedure is repeated until no more valid pairs are available.

4.2 Trajectory Initialization

The basic idea of the initialization strategy is to start a trajectory when enough evidence is collected from the detection responses. Define the precision, pr , of a detector as the ratio between the number of successful detection and the number of all responses. If pr is constant between frames, and the detection in one frame is independent of the neighboring frames, then during consecutive T time steps, the probability that the detector outputs T consecutive false alarms is $P_{FA} = (1 - pr)^T$. However, this inference is

not accurate for real videos, where the inter-frame dependence is large. If the detector outputs a false alarm at a certain position in the first frame, the probability is high that a false alarm will appear around the same position in the next frame. We call this *persistent false alarm* problem. Even here, the real P_{FA} should be an exponentially decreasing function of T , we model it as $e^{-\lambda_{init}\sqrt{T}}$.

Suppose we have found $T(> 1)$ consecutive responses, $\{\mathbf{rc}_1, \dots, \mathbf{rc}_T\}$ corresponding to one object hypothesis H by data association. The confidence of initializing a trajectory for H is then defined by

$$\begin{aligned} & \text{InitConf}(H; \mathbf{rc}_{1..T}) \\ &= \underbrace{\frac{1}{T-1} \sum_{t=1}^{T-1} A(\hat{\mathbf{rc}}_{t+1}, \mathbf{rc}_{t+1})}_{(1)} \cdot \underbrace{\left(1 - e^{-\lambda_{init}\sqrt{T}}\right)}_{(2)} \end{aligned} \quad (4)$$

The first term in the left side of Eq.4 is the average affinity of the T responses, and the second term is based on the detector's accuracy. The more accurate the detector is, the larger the parameter λ_{init} should be. Our trajectory initialization strategy is: if $\text{InitConf}(H)$ is larger than a threshold, θ_{init} , a trajectory is started from H , and H is considered to be a *confident trajectory*; otherwise H is considered to be a *potential trajectory*. A trajectory hypothesis H is represented as a 3-tuple, $\{\{\mathbf{rc}_t\}_{t=1, \dots, T}, \mathbf{D}, \{\mathbf{C}_i\}_{i=FB, HS, TS, L}\}$, where $\{\mathbf{rc}_t\}$ is a series of responses, $\{\mathbf{C}_i\}$ is the appearance model of the parts, and \mathbf{D} is a dynamic model. In practice, \mathbf{C}_i is the average of the appearance models of all detection responses, and \mathbf{D} is modeled by a Kalman filter for constant speed motion.

4.3 Trajectory Growth

After a trajectory is initialized, the object is tracked by two strategies, data association and meanshift tracking. For a new frame, first, for all existing hypotheses, we look for their corresponding responses in this frame. If there is a new response matched with a hypothesis H , then H grows based on data association, otherwise a meanshift tracker is applied. The data association itself has two steps. First, all the hypotheses are matched with the combined responses by the method described in Section 4.1. Second, all hypotheses which are not matched in the first step are associated with the remaining part responses which do not belong to any combined response. Matching part responses with hypotheses is a simplified version of the method for matching combined responses with hypotheses. At least one part must be detected for an object to be tracked by data association.

Whenever data association fails (the detectors can not find the object or the affinity is low), a meanshift tracker [18] is applied to track the parts individually. Then the results are combined to form the final estimation. The basic idea of meanshift is to track a probability distribution.

Although the typical way to use meanshift tracking is to track the color distribution, there is no constraint on the distribution to be used. In our method we combine the appearance model, \mathbf{C} , the dynamic model, \mathbf{D} , and the detection confidence, f , to build a likelihood map which is then fed into the meanshift tracker. A dynamic probability map, $P_{dyn}(\mathbf{u})$, where \mathbf{u} represents the image coordinates, is calculated from the dynamic model \mathbf{D} , see Figure 2(d). Denote the original responses of one part detector at the frame j by $\{\mathbf{rp}_j\}$, the detection probability map $P_{det}(\mathbf{u})$ is defined by

$$P_{det}(\mathbf{u}) = \sum_{j: \mathbf{u} \in \text{Reg}(\mathbf{rp}_j)} f_j + ms \quad (5)$$

where $\text{Reg}(\mathbf{rp}_j)$ is the image region, a rectangle, corresponding to \mathbf{rp}_j , f_j is a real-valued detection confidence of \mathbf{rp}_j , and ms is a constant corresponding to the missing rate (the ratio between the number of missed objects and the total number of objects). ms is calculated after the detectors are learned. Note, the original response is used here, because of possible errors in the clustering algorithm (see Figure 2(e)).

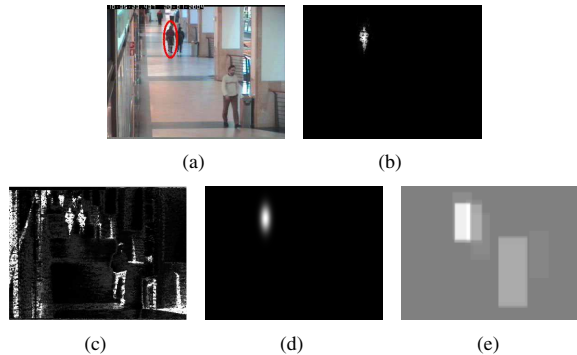


Figure 2. Probability map for meanshift: a) original frame; b) final probability map; c), d) and e) probability maps for appearance, dynamic and detection respectively. (The object concerned is marked by a red ellipse.)

Let $P_{appr}(\mathbf{u})$ be the appearance probability map. As \mathbf{C} is a color histogram, $P_{appr}(\mathbf{u})$ is the bit value of \mathbf{C} (see Figure 2(c)). To estimate \mathbf{C} , we need the object to be segmented so that we know which pixels belong to the object. The the detection response rectangle is not accurate enough for this purpose. Also as a human is a highly articulated object, it's difficult to build a constant segmentation mask. In [19], Zhao and Davis proposed an iterative method for upper body segmentation to verify the detected human hypotheses. Here, we propose a simple PCA based approach. At the training stage, examples are collected and the object regions are labeled by hand, see Figure 3(a). Then a PCA model is learned from this data, see Figure 3(b). Suppose we have an initial appearance model \mathbf{C}_0 . Given a new sample (Figure 3(c)), first we calculated its color probability map from \mathbf{C}_0 (Figure 3(d)), then we use the PCA model

as a global shape constraint by reconstructing the probability map (Figure 3(e)). The thresholded reconstruction map (Figure 3(f)) is taken as the final object segmentation, which is used to update C_0 . The mean vector, the first one of Figure 3(b), is used to compute C_0 the first time. For each part, we learn a PCA model. This segmentation method is far from perfect, but very fast and adequate to update the appearance model.

Combining $P_{appr}(\mathbf{u})$, $P_{dyn}(\mathbf{u})$, and $P_{det}(\mathbf{u})$, we define the image likelihood for a part at pixel \mathbf{u} by

$$L(\mathbf{u}) = P_{appr}(\mathbf{u})P_{dyn}(\mathbf{u})P_{det}(\mathbf{u}) \quad (6)$$

Figure 2 shows an example of probability map computation. Before the meanshift tracker is activated, an inter-object occlusion reasoning is applied. Only the visible parts which were detected in the last successful data association, are tracked. Finally only the models of the parts which are not occluded and detected are updated.

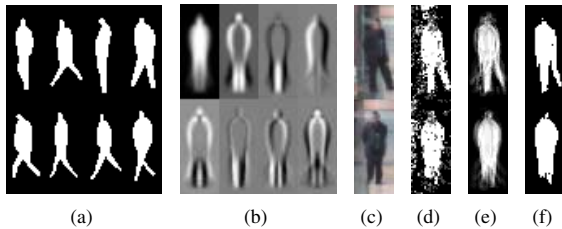


Figure 3. PCA based body part segmentation: a) training samples; b) eigenvectors. The left top one is the mean vector; c) original human samples; d) color probability map; e) PCA reconstruction; f) thresholded segmentation map.

4.4 Trajectory Termination

The strategy of terminating a trajectory is similar to that of initializing it. If no detection responses are found for an object H for consecutive T time steps, we compute a termination confidence of H by

$$\begin{aligned} & \text{EndConf}(H; \mathbf{rc}_{1..T}) \\ &= \left(1 - \frac{1}{T-1} \sum_{t=1}^{T-1} A(\widehat{\mathbf{rc}}_{t+1}, \mathbf{rc}_{t+1})\right) \left(1 - e^{-\lambda_{end}\sqrt{T}}\right) \end{aligned} \quad (7)$$

Note that the combined responses \mathbf{rc}_t are obtained from the meanshift tracker, not from the combined detector. If $\text{EndConf}(H)$ is larger than a threshold, θ_{end} , hypothesis H is terminated; we call it a *dead trajectory*, otherwise we call it an *alive trajectory*.

4.5 Combined Tracker

Now let's put the above three modules, trajectory initialization, tracking, and termination, together. Figure 4 gives our full tracking algorithm. The algorithm is called *forward tracking*, as it only looks ahead. Because the trajectory initialization may have some delay, we also use a *backward tracking* procedure which is the exact reverse of the forward

tracking. After a trajectory is initialized, it may grow in both forward and backward directions.

A simplified version of the combined tracking method is to track only a single part, *e.g.* the full-body. In the results in Section 5.3, we show that the combined tracking outperforms the single part tracking. The combined tracking method is robust because:

1. It uses combined detection responses, whose false alarm rate is low, to start trajectories. However this makes the initialization stage be able to work with inter-object occlusions, but not scene occlusion. One alternative is to start trajectories for part detection responses with the risk of having more false alarms.
2. The combined tracker uses all part responses to track the object. The probability that all part detectors fail at the same time is lower than the probability of a single part detector's failure.
3. In tracking stage, this method can work with both scene and inter-object occlusions, because the object is tracked whenever any part of it is detected.
4. When tracking by meanshift, the combined method takes the average of the tracking results of all parts. If only one part drifts, the object can still be tracked.

Our method doesn't have an explicit strategy to deal with full occlusion, because both the detectors and the trackers rely on 2D observation of the objects. If we can not "see" the object, the dynamic model itself is not strong enough to track the object.

5 Experimental Results

We show results and evaluations on three video sets to demonstrate the effectiveness of our method. The first set is a selection from the CAVIAR video corpus [20], which is captured with a stationary camera, mounted a few meters above the ground and looking down towards a corridor. The frame size is 384×288 and the sampling rate is 25 FPS. The second set, called the "skate board set", is captured from a camera held by a person standing on a moving skate board. The third set, called the "building top set", is captured from a camera held by a person standing on top of a 4-story building looking down towards the ground. The camera motions in the skate board set include both translation and panning, while those of the building top set are mainly panning and zooming. The frame size of these two sets is 720×480 and the sampling rate is 30 FPS. We characterize the occlusion events in these three sets with two criteria: if the occlusion is by a target object, *i.e.* a human, we call it an object occlusion, otherwise a scene occlusion. If the period of the occlusion is longer than 50 frames, it's considered to be a long term occlusion; otherwise a short term one. So we have four categories: short term scene, long term scene, short term object, and long term object occlusions. Table 1 gives the frequencies of the occlusion events in these sets.

Forward Human Tracking

Let the set of hypotheses be F , initially $F = \Phi$.

For each time step t (denote by F_t the set of all alive trajectories in F at time t)

1. Static detection:
 - (a) Detect parts. Let the result set be RP_t .
 - (b) Combine part detection responses, including inter-object occlusion reasoning. Let the result set be RC_t .
 - (c) Subtract the parts used in RC_t from RP_t .
2. Data association:
 - (a) Associate hypotheses in F_t with combined responses in RC_t . Let the set of matched hypotheses be F_{t1} .
 - (b) Associate hypotheses in $F_t - F_{t1}$ with part responses in RP_t . Let the set of matched hypotheses be F_{t2} .
 - (c) Build a new hypothesis H from each unmatched response in RC_t , and add H into F and F_t .
3. Pure tracking: For each confident trajectory in $F_t - F_{t1} - F_{t2}$, grow it by meanshift tracking.
4. Model update:
 - (a) For each hypothesis in $F_{t1} + F_{t2}$, update its appearance model and dynamic model.
 - (b) For each potential trajectory in F_{t1} , update its initialization confidence.
 - (c) For each trajectory in $F_{t1} + F_{t2}$, reset its termination confidence to 0.
 - (d) For each trajectory in $F_t - F_{t1} - F_{t2}$, update its termination confidence.

Output all confident trajectories in F as the final results.

Figure 4. Forward human tracking algorithm.

Video Set		SS	LS	SO	LO	Overall
CAVIAR	Zhao-Nevatia	0/0	0/0	34/66	6/11	40/77
	This method	0/0	0/0	39/66	9/11	48/77
Skate Board		6/7	2/2	11/16	0/0	19/25
Building Top		4/7	11/13	15/18	4/4	34/42

Table 1. Frequencies of and performance on occlusion events. n/m : n successful tracked among m occlusion events. (SS: short scene; LS: long scene; SO: short object; LO: long object)

5.1 Performance Evaluation Criteria

To evaluate the performance of our system quantitatively, we define five criteria for tracking: number of “mostly tracked” trajectories (more than 80% of the trajectory is tracked), number of “mostly lost” trajectories (more than 80% of the trajectory is lost), number of “fragments” of trajectories (a result trajectory which is less than 80% of a ground truth trajectory), number of false trajectories (a result trajectory corresponding to no real object), and the frequency of identity switches (identity exchanges between a pair of result trajectories). Figure 5 illustrates these definitions. These five categories are by no means a complete

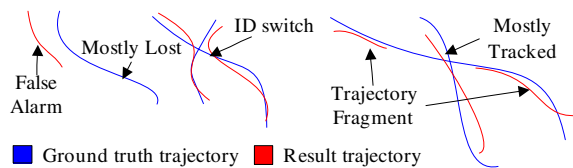


Figure 5. Tracking evaluation criteria.

classification, however they cover most of the typical errors observed in our experiments.

5.2 Results on CAVIAR Set

The only previous method for which we have an implementation in hand is that of Zhao and Nevatia [4]. In this experiment, we compared our method with that in [4]. Their method is based on background subtraction, and requires a calibrated stationary camera. For comparison, we build the first test set from CAVIAR video corpus [20]. This set contains 23 sequences, overall 29,768 frames. The scene is very clear, however the inter-object occlusion is intensive, see Table 1. 19 out of the 77 occlusion events are fully occluded ones (more than 90% part of the object is invisible). Frequent interactions between humans, such as talking, and shaking hands, make this set very difficult for tracking. Table 2 gives the comparative results at tracking level. It can

	GT	MT	ML	Fgmt	FAT	IDS
Zhao-Nevatia	144	93	7	57	20	14
This method		104	6	42	5	13

Table 2. Tracking level comparison with [4] on CAVIAR set, 23 sequences. (GT: ground truth; MT: mostly tracked; ML: mostly lost; Fgmt: trajectory fragment; FAT: false alarm trajectory; IDS: ID switch)

be seen that our method gives many fewer false alarm trajectories than the method of [4], while the other criteria are comparable. This comes from the low false alarm rate of the combined detector. The first two rows of Table 1 give the performance comparison on occlusion events. Tracking success of an occlusion event means that no object is lost,

no trajectory is broken, and no ID switches occur during the occlusion. Our system obtained reasonable performance on partial occlusions. Some sample frames and results are shown in Figure 6.

5.3 Results on Skate Board Set

The main difficulties of the skate board set are small abrupt motions due to the uneven ground, and some occlusions. This set contains 29 sequences, overall 9,537 frames. Only 13 out of them have no occlusion at all. Some sample frames and results are shown in Figure 6. The combined tracking method is applied. Table 3 gives the tracking performance of the system, and the third row of Table 1 gives the performance on occlusion events for this set. It can be seen that our method works reasonably well on this set.

GT	MT	ML	Fgmt	FAT	IDS
50	39	1	16	2	3

Table 3. Performance on skate board set, 29 sequences. (see Table 2 for abbreviations.)

For comparison, a single part (full-body) tracker, which is a simplified version of the combined tracker, is applied on the 13 videos that have no occlusions. Because the part detection does not deal with occlusion explicitly, it is not expected to work on the other 16 sequences. Table 4 shows the comparison results. It can be seen that the combined tracker gives many fewer false alarms than the single part tracker. This is because the full-body detector has more persistent false alarms than the combined detector. Also combined tracker has more fully tracked objects, because it makes use of cues from all parts.

	GT	MT	ML	Fgmt	FAT	IDS
Part Tracking	21	14	2	7	13	3
Combined Tracking		19	1	5	2	2

Table 4. Comparison between part tracker and combined tracker on skate board set, 13 sequences. (see Table 2 for the abbreviations.)

5.4 Results on Building Top Set

The building top set contains 14 sequences, overall 6,038 frames. The main difficulty of this set is due to frequency of occlusions, both scene and object, see Table 1. No single part tracker works well on this set. The combined tracker is applied to this data set. Table 5 gives the tracking performance. The fourth row of Table 1 gives the performance on occlusion events. It can be seen that the combined tracker obtains very few false alarms and a reasonable success rate. Our method can also work in the presence of long term scene or object occlusion. Some sample frames and results are shown in Figure 6.

On all three test sets, the average detection rate of the combined detector is about 55% and there are about 0.1

false alarms per frame. The average detection rate of individual part detectors is 65% and the false alarm rate is 0.36 per frame. We set the parameters to get a low false alarm rate. For tracking, on average, about 50% of the successful tracking is due to the data association with combined responses, *i.e.* the object is “seen” by the combined detector; about 35% is due to the data association with part responses; the remaining 15% is from the meanshift tracker. Although the detection rate of any single detector is not high, the tracking level performance of the combined tracker is much better. our combined tracker can get good performance on tracking level. The speed of the entire system is about 0.1 FPS on a 2.8G Hz Pentium CPU; the program is coded in C++ using OpenCV functions. Most of the computation cost is in the static detection component.

GT	MT	ML	Fgmt	FAT	IDS
40	34	3	3	2	2

Table 5. Performance on building top set, 14 sequences. (see Table 2 for the abbreviations.)

6 Conclusion and Future Work

We proposed a fully automatic human tracking method based on body part representation. The responses of static combined human detection and body part detection are taken as the observations of the human hypotheses and fed into the tracker. Both the trajectory initialization and termination are based on the evidence collected from the detection responses. To track the objects, most of the time data association works, while a meanshift tracker fills in the gaps between data association. From the experimental results, it can be seen that the proposed system has low false alarm rate and achieves high successful tracking rate. It can work under both partial scene and inter-object occlusion conditions reasonably well.

The comparison with the method in [4] is done on a case where both methods work. However, each has its own limits. The method of [4], which is based on 3D model and motion segmentation, is less view dependent and can work on lower resolution videos, while our method, which is based on 2D shape, requires a relative high resolution and does not work with a large camera tilt angle. On the other hand, our method, which is based on frame by frame detection, can work with moving and/or zooming cameras, while the method of [4] can not.

Currently our system does not make use of any cues from motion segmentation. When motion information is available, it should help improve the performance. For example, recently Brostow and Cipolla [21] proposed a method to detect independent motions in crowds. The outputs are *tracklets* of independently moving entities, which may facilitate object level tracking. Conversely, shape-based tracking can help improve motion segmentation. We plan to explore such interactions in future work.

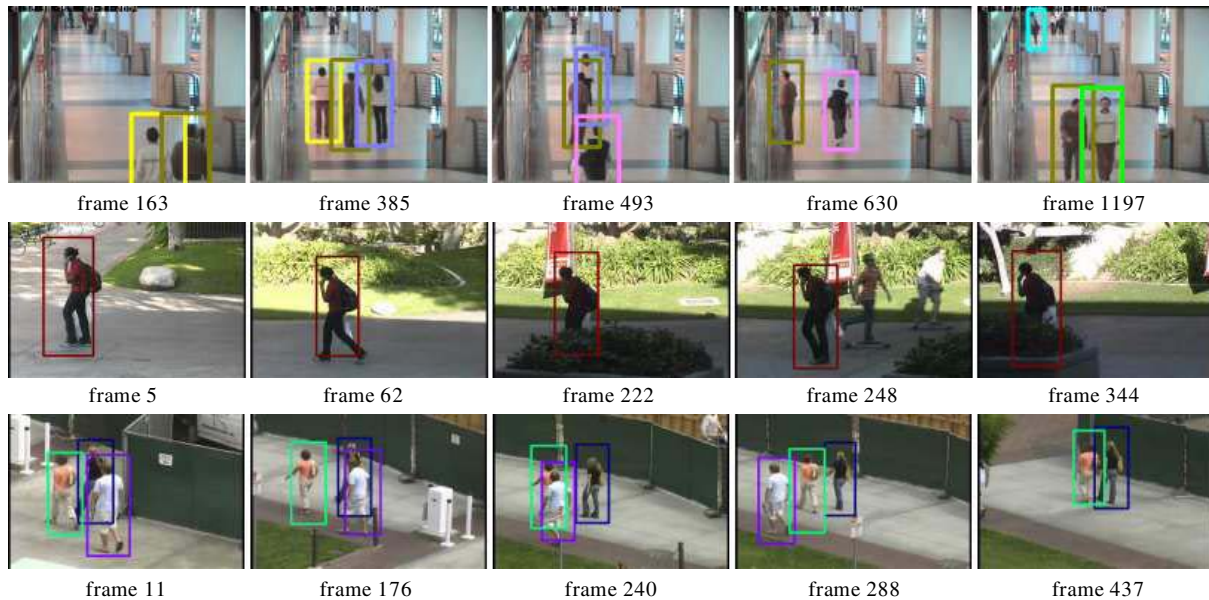


Figure 6. Sample tracking results. The first row is from CAVIAR set; the second row is from skate board set; the third row is from building top set.

Acknowledgements: The authors would like to thank Tae Eun Choe and Qian Yu for their help for capturing the videos. This research was partially funded by the Advanced Research and Development Activity of the U.S. Government under contract MDA-904-03-C-1786.

References

- [1] B. Wu, and R. Nevatia. Detection of Multiple, Partially Occluded Humans in a Single Image by Bayesian Combination of Edgelet Part Detectors. ICCV'05. Vol I: 90-97
- [2] C. Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a probabilistic assembly of robust part detectors. ECCV'04. Vol I: 69-82
- [3] A. Mohan, C. Papageorgiou, and T. Poggio. Example-based object detection in images by components. Trans. PAMI, 23(4):349C361, 2001.
- [4] T. Zhao and R. Nevatia. Tracking multiple humans in crowded environment. CVPR'04. Vol II: 406-413
- [5] L. Sigal, S. Bhatia, S. Roth, M. J. Black, M. Isard. Tracking Loose-limbed People. CVPR'04. Vol I: 421-428
- [6] M. Lee, and R. Nevatia. Human Pose Tracking using Multi-level Structured Models. ECCV'06.
- [7] B. Leibe, E. Seemann, and B. Schiele. Pedestrian Detection in Crowded Scenes. CVPR'05. Vol I: 878-885
- [8] C. Papageorgiou, T. Evgeniou, and T. Poggio. A Trainable Pedestrian Detection System. In: Proc. of Intelligent Vehicles, 1998. pp. 241-246
- [9] P. Felzenszwalb. Learning models for object recognition. CVPR'01. Vol I: 56-62
- [10] D. Gavrilu. Pedestrian detection from a moving vehicle. ECCV'00. Vol II: 37-49
- [11] P. Viola, M. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. ICCV'03. pp. 734-741
- [12] Y. Wu and T. Yu and G. Hua. A Statistical Field Model for Pedestrian Detection. CVPR'05. Vol I: 1023-1030
- [13] K. Smith, D. G.-Perez, and J.-M. Odobez. Using Particles to Track Varying Numbers of Interacting People. CVPR'05. Vol I: 962-969
- [14] J. R. Peter, H. Tu and N. Krahnstoever. Simultaneous Estimation of Segmentation and Shape. CVPR'05. Vol II: 486-493
- [15] M. Isard and J. MacCormick. BraMBLE: A Bayesian Multiple-Blob Tracker. ICCV'01. Vol II: 34-41
- [16] L. Davis, V. Philomin and R. Duraiswami. Tracking humans from a moving platform ICPR'00. Vol IV: 171-178
- [17] P. Viola and M. Jones. Rapid Object Detection using a Boosted Cascade of Simple Features. CVPR'01. Vol I: 511-518
- [18] D. Comaniciu, V. Ramesh, and P. Meer. The Variable Bandwidth Mean Shift and Data-Driven Scale Selection. ICCV'01. Vol I: 438-445
- [19] L. Zhao, L. Davis. Closely Coupled Object Detection and Segmentation. ICCV'05. Vol I: 454-461
- [20] <http://homepages.inf.ed.ac.uk/rbf/CAVIAR/>
- [21] G. J. Brostow, and R. Cipolla. Unsupervised Bayesian Detection of Independent Motion in Crowds. CVPR'06.