Abstract One of the main difficulties in video tracking of people arises in scenarios where targets are repeatedly and extensively occluded by other moving objects. These types of occlusions significantly affect the measurements of the person's position, motion, shape and appearance, posing major challenges to correct tracking and data association. In this paper, we present a method for tracking people in videos based on a simplified part-based model only loosely associated with body parts. Data association is provided by a layered data association approach which performs association at feature, part and global levels in a hierarchical fashion. Occlusions are detected and managed at the part level, with corresponding model update strategies. In addition, the tracker does not make any assumption on the target's motion direction, thus allowing tracking to withstand abrupt sideways movements and changes of directions that frequently occur in busy scenes. Experimental results against popular trackers such as mean shift, particle filters and the recent k-shortest paths (KSP) tracker based on a variety of performance indicators and datasets including ETISEO, AVSS 2007 and PETS 2009 show the effectiveness of the proposed tracker.

Keywords Video Tracking · Layered Data Association · Tracking Under Occlusions · Part-Based Models

Multimedia Tools and Applications manuscript No.
(will be inserted by the editor)

Tracking People under Heavy Occlusions by Layered Data Association

Zui Zhang · Oscar Perez Concha · Massimo Piccardi

Received: / Accepted: date

1 Introduction and Related Work

People tracking is a critical component of many computer vision applications such as surveillance, human-computer interaction, media annotation and several others. However, tracking in visual data is intrinsically challenged by view occlusions obscuring the target from the camera. The view occlusion problem is especially serious in situations where multiple targets are present at once such as in public environments: pedestrians repeatedly occlude each other either partially or completely as they walk or stand in the area, form groups, stop to interact with others or simply are temporarily occluded by the infrastructure in the scene. Almost invariably, this situation leads to the eventual loss of the target. Despite being the focus of recent research ([2,11,19,24,21,18], amongst others), tracking people under repeated and substantial occlusions is still a partially unresolved problem in computer vision.

The question we address in this paper is whether a simplified representation of the target can provide an adequate basis for tracking humans under frequent and extensive occlusions such as those occurring in moderately crowded environments. In our reference scenarios, humans are typically 50 to 200 pixels in height, are occluded repeatedly and extensively by other walking or standing humans, and change directions often and unpredictably in order to avoid collisions. These are the typical viewing conditions of wide-area surveillance cameras in shopping centers, train stations, airports, with usual frame rates in the order of 15 to 30 frames per second.

To this aim, in this paper we propose a target model based on parts loosely associated with the human's head, left torso and arm (with respect to the viewpoint), right torso and arm, left leg, and right leg. Each part is represented by a rich set of features including the part's color histograms and shape descrip-

University of Technology, Sydney (UTS), PO Box 123, Broadway, NSW, Australia
Corresponding author: Massimo Piccardi
email: Massimo.Piccardi@uts.edu.au; phone: +61 2 95147942; fax: +61 2 95144535

tors. Data association between the target model and observations in successive frames is based on a *layered data association* approach which performs association at feature, part and global levels in a hierarchical fashion. Occlusions are detected and managed at the part level: whenever a major occlusion is detected over a model's part, its model is kept unchanged until the occlusion has ceased. Conversely, the models of unoccluded parts are updated meanwhile. In addition, the tracking algorithm does not make any assumption on the target's motion direction, only searching for the best candidate within an adjustable spatial window in order to withstand sudden changes in direction.

Part-based approaches are becoming increasingly popular for detection of deformable objects thanks to their intrinsic ability to adjust to deformations. In [4], Felzenszwalb *et al.* applied a part-based model for the detection of objects subject to major deformations (i.e. humans). In [12,11], the authors have proposed a hierarchical part-template matching approach to simultaneous human detection and segmentation, and its integration with a tracker [19] for tracking purposes. The main difference with our approach is that we do not seek accurate segmentation of the tracked humans at any stage and can therefore rely on algorithms which are simpler and faster in principle. Zhao *et al.* in [24] and Wu and Nevatia in [21] have also proposed approaches for part-based human tracking. The main difference of the method proposed in this paper is that it always imposes a constraint of global integrity to the matching of the individual parts. In other words, the proposed method only detects humans globally, not parts individually. The most recent paper on part-based tracking we are aware of was presented by Shu *et al.* in 2012 [18]. Their method adopts a sophisticated, 8-part model requiring high-resolution videos (in the order of 1920×1080 frame size) to be fitted effectively. Conversely, our method is based on a simpler model designed to work with the the low-medium frame resolution typical of commercial surveillance systems.

The main contributions of the proposed approach, simply called *Part-Based Model* (PBM) hereafter, are summarized as follows:

- the use of a human model striking an effective trade off between complexity and fitting feasibility in medium-resolution videos typical of wide-area surveillance cameras;
- the adoption of a layered data association approach allowing tracking to withstand the occlusions common in moderately crowded scenarios such as shopping centers, train stations, airports and other public premises. This makes the proposed approach widely applicable;
- a strong experimental performance against popular trackers such as mean shift [16], particle filters [14,17] and the recent k-shortest paths optimization [2] over a variety of performance indicators and datasets including ETISEO [13], AVSS 2007 [1] and PETS 2009 [15].

Despite its use of parts to provide data association, this model should not be confused with approaches to human articulated motion tracking (see [9] for a reference). In articulated motion tracking, the objective is to explicitly track the human's limbs and articulated degrees of freedom whereas in our approach the goal is just that of tracking the human as a global entity. In general, articulated motion tracking requires closer views than those typical of wide-area surveillance cameras.

2 Part-Based Model for Tracking

The framework of the proposed approach consists of a) an adaptive *global model* containing the target's location, width, height and centroid which is updated at all frames; b) an adaptive *parts model* made of five body parts, each updated if not occluded; c) a velocity model which assumes constant velocity magnitude but no direction, and d) an algorithm for layered data association.

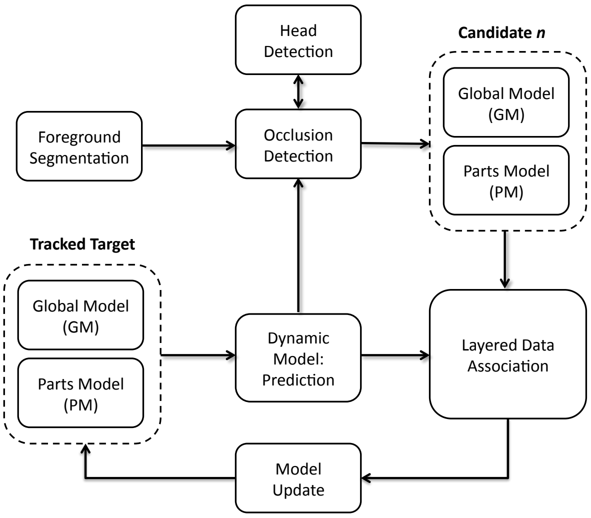


Fig. 1 Flow diagram of the proposed tracker.

Figure 1 shows the workflow of our tracking algorithm. The main steps can be described as follows:

- **Foreground segmentation:** The foreground areas in each frame are extracted by background subtraction, using a Gaussian mixture background model [6]. Subsequently, we apply morphological operations and shadow removal to improve the quality of the detection. We care to note that this step is not required for the operation of the tracker, but speeds it up significantly by focussing only on relevant regions.

- **Dynamical model: prediction:** the model for the tracked target (global and parts models, Sections 2.1 and 2.2, respectively) is predicted by way of the dynamical model (Section 3).
- **Occlusion detection:** occlusions are detected by comparing the target's predicted model and a candidate's model (Section 3.1). In case of occlusions, we align the predicted model to the candidate based on a separate **head detection** procedure (Section 2.3).
- **Layered data association (Section 4):** at each frame, the comparison between the target's predicted model and a candidate's model is carried out through a layered data association which includes features, parts and global layers.
- **Model update (Section 4.2):** upon a match, the target's global model is updated; its corresponding parts model is updated for the unoccluded body parts.

2.1 The global model

The global model (GM) consists of five rectangles in approximate correspondence with the head, left and right torso areas, and legs. The GM has an overall rectangular shape and six degrees of freedom (DOF) in the image plane, namely: the coordinates of the top left corner, (x_{tl}, y_{tl}); the centroid's coordinates, (x_c, y_c); and the bounding box' width and height, (w, h). The anatomy of the human body makes it possible to assume that the head occupies a fixed ratio (one-seventh) of the person's height. Such assumptions split the overall rectangle into five sub-rectangles as shown in Fig. 2. During motion, the torso regions provide some desirable stability to the overall model.

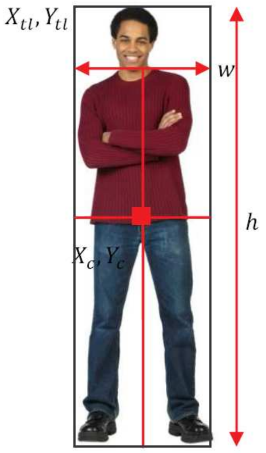


Fig. 2 The global model (GM).

A global model is maintained for the target at all frames, recording its last validated position. A global model is also laid out over each "candidate blob" to perform data association. The construction of the GM for each candidate blob differs depending on the candidate blob's status. If the candidate blob is unoccluded, the six DOF are measured directly from the observations: (x_c, y_c) is computed as the blob's centroid; (w, h) are measured as the blob's horizontal and vertical span; x_{tl} and y_{tl} are measured from the left-most and top-most pixels of the blob, respectively. However, if the candidate blob contains major occlusions, it is not possible to directly measure all the six DOF. In this case, the model is estimated based on the results of a head detector:

- Coordinates x_c and y_{tl} are taken from the top-most pixel of the detected head.
- The width, w, and height, h, are estimated using the Kalman filter and the search procedure described in section 3.
- The relative location of the centroid proves rather stable within the human body despite deformations. We thus compute two ratios, α and β, that represent the relative position of the centroid within the bounding box. Both ratios are updated by a running average with a window-size of a few frames (3 to 5). The first ratio, α, is the ratio between the width of the left-hand side body part and the total width:

  $$\alpha(n) = \frac{x_c(n) - x_{tl}(n)}{w(n)} \tag{1}$$

  whereas the second ratio, β, is the ratio between the height of the torso and the total body height:

  $$\beta(n) = \frac{y_{tl}(n) - y_c(n) - \frac{1}{7}h(n)}{h(n) - \frac{1}{7}h(n)} \tag{2}$$

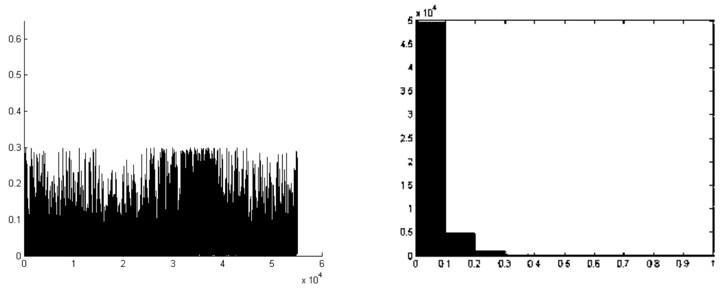
In the case of occlusions, we use the values of α and β at frame $n-1$ to infer x_{tl} from (1) and y_c from (2).

2.2 The parts model

The parts model (PM) is the model of all the body parts. Each such a part is modelled by a set of N_f features, $\{F_i\}, i = 1...N_f$, which were selected following empirical criteria such as limited variance to pose changes, deformations, illumination variations, and mild occlusions. The feature set includes:

- **Area**: number of foreground pixels within the body part. This feature is relatively invariant to both illumination changes and body part deformations. During light occlusions, this feature is most likely to remain stable; however, if the body part is significantly occluded, it will vary significantly.
- **Perimeter**: length of the contour of the foreground region that represents the body part. This feature, too, is relatively invariant to both illumination changes and body part deformations.

- **Color histogram**: color histogram of the body part in both HSV and XYZ color spaces. Color histograms are essentially invariant to deformations. However, they are very vulnerable to sharp illumination changes and major occlusions.
- **Displacement**: displacement between the centroid of the same body part between consecutive frames. The values are expected to be bounded by physical motion, whereas major occlusions may cause unexpectedly large values.
- **Amount of overlap**: amount of overlap between consecutive frames. It is expected to be somehow stable if the part is in full view or subject to minor occlusions or deformations, but likely to vary abruptly during major occlusions.



(a) (b)

Fig. 3 (a) Absolute area differences for the left arm body part over a collection of frames from various video datasets [13,15,1] (b) Histogram of the absolute area differences (figure 3(b)).

The main goal of these features is to support correct data association throughout tracking. As such, they must prove discriminative over the two basic cases of "matching with the correct target" and "non matching". Modelling statistically the non-matching case is hard as the combinations are too varied. Instead, modelling the matching case is possible by identifying the features' expected range of variations over single targets. To this aim, we conducted a major preliminary experiment by manually annotating the target and its parts in each frame of several videos from various datasets [13,15,1]. In detail, we have manually annotated 20 video sequences over different scenarios such as a corridor, a train station, a shopping mall, and various outdoor scenes, each comprising between 1,000 and 4,000 frames [13,15,1] for a total of over 54,000 frames. For each frame, we have computed the absolute difference between the features in the frame and those in the previous frame, for corresponding parts, and modelled such differences by half-normal distributions. As an example, Fig. 3(a) shows the differences for feature *area* for the left arm part over the entire collection of frames: the distribution is clearly unimodal and short-tailed and suitable to be modeled as a half-normal distribution. Where unimodality was not evident from the histogram, we have used Gaussian mixtures for modelling. Using our annotated training set, it would also be possible to add other features to the parts models, including popular local descriptors such as the histograms of oriented gradients and the histograms of optical flow [10].

2.3 Head detection

The head detector used in this work uses a combination of hair and skin's appearance models and an elliptical shape constraint based on the human anatomy. For a broader evaluation of its performance, the reader may refer to [23].

2.3.1 Hair color model

From a qualitative analysis, human hair colors can be seemingly clustered into a few basic color categories, namely black, blond, red and brown. Each category is here represented by a probability density function, in particular, a Gaussian mixture model (GMM) whose parameters are fitted by an expectation-maximization (EM) algorithm [3].

For a pixel to be labelled as a hair pixel, the likelihood of its color in the hair distribution models must be high. Let us assume that a pixel, x, is represented as $\{x_X, x_Y, x_Z\}$ and $\{x_H, x_S, x_V\}$ in the XYZ and HSV color spaces, respectively, and that H is a variable representing the hair color category, $H \in \{\text{black, blond, brown, red}\}$. The likelihoods of x given H in the XYZ and HSV color spaces are then estimated as follows:

$$p^{XYZ}(x|H) = p(x_X|H) \times p(x_Y|H) \times p(x_Z|H) \tag{3}$$

$$p^{HSV}(x|H) = p(x_H|H) \times p(x_S|H) \times p(x_V|H). \tag{4}$$

The right members of (3) and (4) are a simplification of the joint probability assuming statistical independence between color channels. The combined likelihood of x given H for the two color spaces is then estimated as:

$$p_{hair}(x|H) = w_1 p^{XYZ}(x|H) + w_2 p^{HSV}(x|H) \tag{5}$$

In an ideal case, $p^{XYZ}(x|H)$ and $p^{HSV}(x|H)$ should have very similar values; therefore, an assumption of independence in (5) would be inappropriate and the combined likelihood is instead approximated by a weighted sum criterion. The final likelihood of pixel x being a hair pixel, $p_{hair}(x)$ is estimated as:

$$p_{hair}(x) = \max_H (p_{hair}(x|H)) \tag{6}$$

In (6), we deliberately avoid the use of priors over the categories to not trade off individual accuracy with higher accuracy over the entire population.

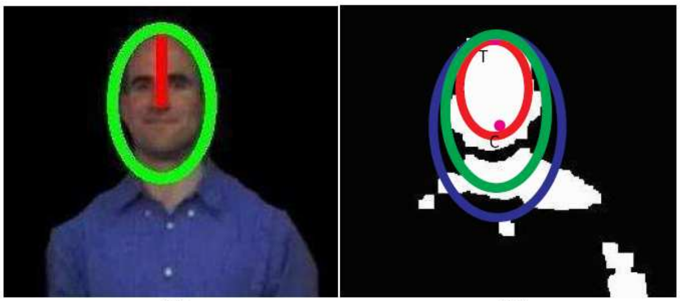
Eventually, a binary decision is made for x based on inequality $p_{hair}(x) > th_{hair}$, where th_{hair} is a threshold determined experimentally. Weights in (5) are chosen by a search over interval [0,1] in steps of 0.1 units so as to maximize correct hair pixel detection over a training set.

2.3.2 Skin color model

Skin colors are distributed differently from hair colors, spreading more continuously, yet within a bounded range [5]. As we found that the illumination conditions have a greater influence on detection of skin than they do on hair, we decided to model the skin colors based on three illumination categories, namely, bright, standard, and dark. A Gaussian mixture is fitted based on training data for each category in both XYZ and HSV color spaces. The model's equations are identical to those used for the hair colors and thus not repeated here.

2.3.3 Shape constraints

After detection of pixels from either the hair or skin model, a morphological closure is first applied to the resulting image. Then, a set of ellipses is fitted in order to find the best-fitting ellipse: the fitting is registered to the top position of the image (see Fig. 4(b)) and the ellipse with highest occupancy ratio is chosen to represent the detected head.



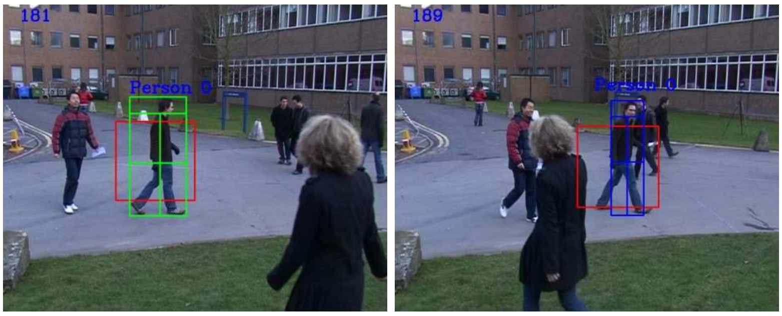
(a) (b)

Fig. 4 Illustration of the head shape modelling and fitting procedure.

The head detector presented in this section is essentially designed to detect uncovered heads. For tracking, the targets' heads do not need to be detected in all frames thanks to the predictive capability of the Kalman filters and the robust data association. Therefore, transient and localized lighting changes and occlusions could be well tolerated. However, detection will likely fail in the presence of head coverings such as hats and veils. For such cases, it could be possible to integrate this detector with a more generic detector such as [20].

3 Dynamical model

The dynamical model of our tracker is based on two dedicated Kalman filters. They are used in two distinct processes, namely *gating* and *target's size*. The gate is defined as the rectangular region in which the target is expected to appear in the current frame given its location in the previous frame (see Fig. 5). A suitable gate estimate proves fundamental for correct data association and for limiting the computational load. We assume the direction of human movement to be unpredictable (sudden changes of direction can occur due to interference with other targets), while speed magnitude is approximately constant because of the intentionality of motion. Therefore, the gate's center is assumed to be the target's location in the previous frame; this equates to a zero-order prediction of the location. The gate's size in both directions is assumed proportional to the target's speed magnitude which, in turn, is estimated by a constant-velocity Kalman filter. The target's measurement vector and state vector are based on its centroid's coordinates as follows:



(a) (b)

Fig. 5 Gating (wider rectangle): (a) normal event; (b) occlusion.

$$KF_1 : Z_1 = \{x_c, x_c\}; \quad X_1 = \{x_c, y_c, \dot{x}_c, \dot{x}_c\} \tag{7}$$

In this way, this Kalman filter simulates a *random walk*, but the target's speed can also be kept updated by way of suitable filter's parameters. In the case the target was not successfully tracked in the previous frame the gate's size is doubled, and enlarged accordingly in the following frames until either the target is tracked again or eventually lost. A blob is considered within the gate if its centroid falls within the gate boundaries.

Prediction is also applied to estimate the size, (h, w), of the global model in the current frame. This prediction is, too, carried out by way of a Kalman filter of measurement vector and state vector:

$$KF_2 : Z_2 = \{h, w\}; \quad X_2 = \{h, w, \dot{h}, \dot{w}\} \tag{8}$$

The estimation of the target's current size is critical as it supports detection of occlusions: a foreground occlusion typically results in a merged blob significantly larger than the target's predicted size. Any background occlusion, conversely, leads to a blob of smaller size.

In all the experiments reported in this paper, the target and its initial position were selected manually in order to study the performance of tracking over a variety of cases of interest. For selection and initialization, the user just drew a bounding box around the chosen target. At run time, this procedure can be replaced by any usual heuristics such as the first appearance of a new blob or the detection of a split event [22].

3.1 Occlusion detection and management

For tracking and data association purposes, the algorithm needs to establish when a foreground blob is occluded. To this aim, an occlusion is declared if the difference between the target's predicted width and height and the candidate blob's width and height are above a significant threshold (set to 50% in either increasing or decreasing direction). This simple rule covers all the cases of merging blobs, overlapping blobs, major segmentation errors and occlusions due to the background scene.

If the blob's size is larger than the target's predicted size, the target needs to be located within the blob. In this situation, our core assumption is that the tops of the heads are visible most of the time. Such an assumption is widely utilized in the tracking literature [22]. Therefore, we use the head as an anchor for aligning our human model onto candidate regions inside the blob: for each detected head in the blob, we generate a set of widths and heights, $(\widetilde{w}_i, \widetilde{h}_i)$:

$$\widetilde{w}_i = w + \delta \times i, \quad i = -2, -1, 0, 1, 2 \tag{9}$$

$$\widetilde{h}_i = h + \delta \times i, \quad i = -2, -1, 0, 1, 2 \tag{10}$$

where δ is fixed to 0.05 and w and h are the predicted width and height of the target in the current frame, respectively, which we use to carve a region inside the blob from the top of the head, down. For each width and height pair, we compare a matching score between the region and the target. Amongst the set of widths and heights, we choose those returning the highest matching score. Empirically we found that the use of heads as anchors and these basic assignment rules between models and blobs provide equivalent information to depth ordering in most cases.

If the blob's size is smaller than the target's predicted size, we look for a head in the blob and attempt alignment with the target's model. If detection or alignment fails, the candidate is dismissed.

4 Layered data association

In the proposed tracker, data association is performed with a layered approach. The matching between the target and a candidate is referred to as *global match* whereas the matching between corresponding body parts of the target model with the candidate model is referred to as *local match*. Local matching is based on direct feature comparison while the global match is inferred from the results of the individual local matches.

The overall matching process is described by the following steps:

1. For each candidate, divide its blob region into five parts as described in Section 2.1. Extract features for each part.
2. Apply part-by-part feature comparison between the target's parts model and the candidate's parts to calculate local match scores.
3. Infer the global match based on the scores of the local matches.
4. Choose the candidate providing the highest global match score. This score must also be above an assigned threshold and provide an adequate ratio against the runner-up candidate (if any).
5. If multiple, nearby candidates result in similar global match scores, apply multiple-response pruning.
6. Update the target's model based on the selected candidate and a part-by-part feature update scheme.

The matching and updating processes are further described hereafter.

4.1 Local and global matches

For each body part, the difference between each of its features for the candidate and the target's models is computed as follows:

$$d_{ij} = |f_{ij} - F_{ij}|, i = 1...N_f, j = 1...N_p \tag{11}$$

where f_{ij} is the value of feature i of part j and F_{ij} is the corresponding feature for the target's model. The number of parts is set to N_p and the number of features, equal for each part, to N_f. In order to determine the probability of matching for d_{ij}, we make use of the $p(d_{ij}|\theta_{ij})$ model of the ground-truth differences between successive target's values described in section 2.2; θ_{ij} are the model's parameters. The part match score is then computed as:

$$p_j = \sum_{i=1}^{N_f} \omega_i p(d_{ij}|\theta_{ij}), j = 1...N_p \tag{12}$$

according to a weighted average fusion rule [8]. The weights in (12), ω_i, have been trained with maximum cross-validation accuracy by a search over interval [0,1] and normalized to add up to 1.

The global match score is eventually inferred as the weighted average of the part match scores:

$$p = \sum_{j=1}^{N_p} \gamma_j p(j) \tag{13}$$

The weights, γ_j, have been set empirically to reflect our prior belief in the stability of the features and the probability of occlusions over the different body parts.

The candidate with the highest (and sufficient) probability is eventually flagged as the current position of the target. However, if multiple potential candidates in nearby locations have similar global match scores or all their global match scores are weak, a final decision is not made at this stage. Rather, *multiple-response pruning* is applied as follows: the most likely reason for having multiple responses is the presence of several misaligned candidates built around the target, typically due to inaccurate head detection nearby the location of the target's head. As correction, an image template of the target's head is exhaustively matched in a local neighborhood until a refined position for the head is found. The matching candidate is then updated accordingly.

4.2 Model update

Updates are performed at both the global and local levels. Update at the global level requires updating the six DOF's of the global model. If the candidate is unoccluded, such DOF's are simply replaced by those of the matching candidate. In the presence of occlusions, the six DOF's are not completely replaced by the candidate's. Rather, an average over a sliding temporal window is used to update their values. The parts' update instead depends on the part's match score. For each part, a decision is made as whether the part matches or not based on score thresholding of (12). If the part matches, a partial update of its area, perimeter, color histogram, and amount of overlap is performed by using a running average. The update weight for each feature is based on its matching probability: the higher the matching probability, the higher is the update weight for the measurement, and vice versa. In the case of "no match" all features remain unchanged in the model.

5 Experimental results

In this section, we report the performance of the proposed tracker over three diverse and challenging datasets and compare it with that of three tracking algorithms representative of the state of the art for various tracking approaches: the mean-shift tracker (MS), representative of appearance-based trackers [16]; the connected-component mean-shift with particle filter tracker (CCMSPF), representative of particle filters [14,17]; and the multiple-object tracker using k-shortest paths optimization (KSP), representative of trackers optimizing data association over multiple frames [2]. At the time of conducting these experiments, we did not have access to other part-based trackers and only a qualitative comparison is addressed in this section. Moreover, since the KSP tracker requires a calibration stage, it had to be simulated "a posteriori" from the elements of the scene; this was only possible for one of the datasets (i.e., ETISEO).

5.1 Video sets

The first video set is the *PETS 2009 sparse crowd people tracking dataset* which contains multiple videos of an outdoor scene captured from different locations at different times [15]. The video used in our experiment was captured with a stationary camera looking toward a T-intersection pathway in a campus (6, left column). The frame size is 720 x 576 and the sampling rate is 25 fps. The video contains overall 939 frames and 26 targets causing frequent and extensive mutual occlusions. In addition, the pedestrians in this video walk in a very unpredictable way: moving backwards, with sudden turns, uneven motions, 'u' turns and 's' shape trajectories.

The second video set, called *AVSS 2007 i-LIDS abandoned baggage detection*, is taken as a part of CCTV surveillance footage from a railway station platform [1]. The frame size is 720 x 576 with a sampling rate of 25 fps. The video selected for the experiment is classified as of medium crowdedness, with prolonged and heavy occlusions and overall 4,833 frames (see Fig. 6, mid column). We have selected three targets to be tracked for the experiment, each of which contains a long-term occlusion.

The third video set is called *ETISEO* [13]. Amongst all the videos, we selected the video recorded in the central hall of a metro station for its challenging occlusions (see Fig. 6, right column). The frame size is again 720 x 576 and the sampling rate 25 fps. In addition, the sequence is of very poor video quality. There is only one target that remains in the scene for the entire sequence. This target is walking around the center of the lobby, with frequent changes of pose and direction, and is also frequently occluded by passing pedestrians. This makes it an ideal choice for comparing tracking performance in an extremely challenging environment. To provide calibration for the KSP tracker, we exploited the regular tiling pattern visible in Fig. 6.

5.2 Performance evaluation: accuracy analysis at the trajectory level

Performance evaluation of trackers often utilize the popular CLEAR's MOTA and MOTP metrics [7]. These metrics compound the tracking accuracy in a single figure and are useful for comparison across the literature. However, in this paper we choose to provide a more comprehensive analysis to better understand the potential and failure of the compared trackers. To measure tracking accuracy on a trajectory basis, we define the following criteria:

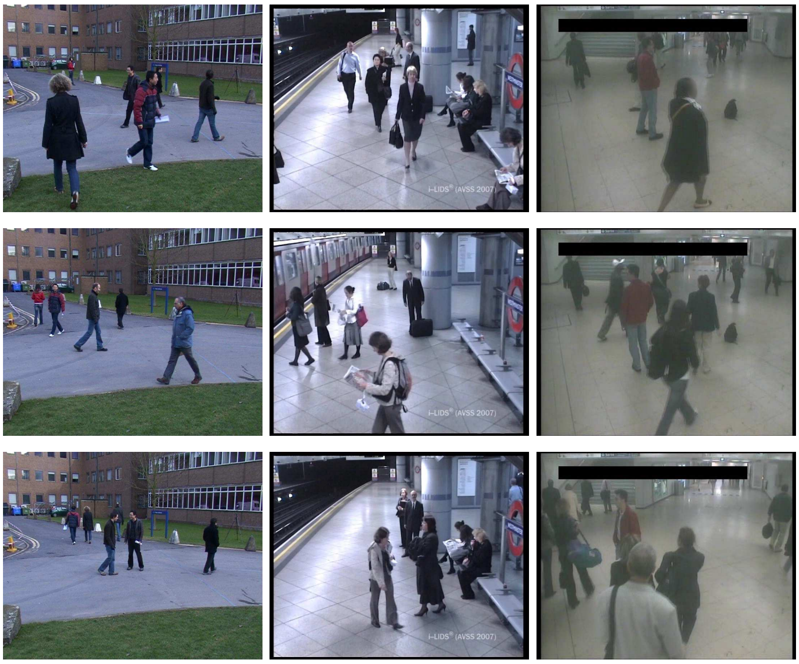


Fig. 6 Example frames from the three video sets. Left column: PETS 2009; mid column: AVSS 2007; right column: ETISEO.

- the number of *mostly-tracked trajectories* (tracked frames greater than 80%)
- the number of *mostly-tracked trajectories with a single ID* (tracked frames greater than 80% and a single ID assigned to the target over the entire trajectory)
- the number of *mostly-lost trajectories* (tracked frames less than 40%)
- the number of *missed trajectories* (trajectory being completely missed)
- the number of *under-segmented trajectories* (target being identified as an existing trajectory rather than initialized with a new ID)
- the number of *over-segmented trajectories* (multiple IDs assigned to a single trajectory)
- the frequency of *identity switches* (ID swapping between two trajectories during their intersection; should be zero)
- the average number of *unique IDs per trajectory* (should be one)
- the average number of *unique IDs per frame* (should be one)

For this type of analysis, we used a set of trajectories from the PETS 2009 video set involving challenging occlusions and sudden changes in direction. Table 1 reports the results: the performance of the PBM tracker is significantly better than that of both the MS and CCMSPF trackers for all criteria. The CCMSPF tracker, at its turn, proves more accurate than the MS tracker. We believe that the main reason for the good performance of PBM is its lack of assumptions on the direction of motion which helps it withstand sharp direction changes.

Table 1 Comparison between different trackers (challenging trajectories) for sequences from PETS 2009.

PETS 2009	GT	PBM	MS	CCMSPF
No. of mostly-tracked trajectories	10	10	8	5
No. of mostly-tracked trajectories (single ID)	10	10	0	2
No. of mostly-lost trajectories	0	0	0	0
No. of missed trajectories	0	0	1	1
No. of under-segmented trajectories	0	0	1	0
No. of over-segmented trajectories	0	0	5	3
Average freq. of ID switches per trajectory	0	0	3.11	1.44
Average no. of unique IDs per trajectory	1	1	3	2.33
Average no. of unique ID per frame	1	1	1.61	1

5.3 Segmentation accuracy

To evaluate the segmentation accuracy, we define four basic metrics for the relative difference between the position of the target's centroid as determined by a tracker, (x_c, y_c), and that of the ground truth, (x_g, y_g), and the relative difference between the size of the target's bounding box as determined by a tracker, (w, h), and that of the ground truth, (w_g, h_g). All differences are averaged over multiple trajectories, multiple frames and the possibly multiple responses of a tracker in each single frame.

Table 2 reports the differences for the PETS 2009 tracks. The table shows a major difference in segmentation accuracy between the results of the PBM tracker and those of the compared trackers. PBM is capable of tracking the target accurately with an average difference of 3.0% and 1.8% for the centroid's position and 13.1% and 5.2% for the bounding box' size. For the MS and CCMSPF trackers, such differences are much larger.

Table 2 Segmentation accuracy of the different trackers (PETS 2009)

	PBM	MS	CCMSPF
x_c difference	3.0%	5.1%	14.2%
y_c difference	1.8%	21.0%	23.1%
Width difference	13.1%	25.9%	23.1%
Height difference	5.2%	24.8%	20.1%

Table 3 reports the differences for the AVSS 2007 video. In this video, the differences in accuracy between PBM and the other two trackers are even more remarked, especially in the estimation of the the bounding box' size which reaches errors of 32.8% and 44.8% with CCMSPF and MS compared to 11.2% for PBM.

Table 3 Segmentation accuracy of the different trackers (AVSS 2007)

	PBM	MS	CCMSPF
x_c difference	2.5%	12.3%	17.8%
y_c difference	2.8%	17.2%	24.2%
Width difference	11.2%	36.5%	33.5%
Height difference	8.0%	44.8%	32.8%

In the more challenging ETISEO video, the MS and CCMSPF trackers substantially fail to track the target after the initial frames as the occlusions become more severe. This is given evidence in Table 4 where the errors from these two trackers reach values of 48.6% and 66.0%. Instead, the PBM tracker achieves errors comparable to those of the simpler PETS 2009 and AVSS 2007 videos. The KSP tracker reports a better performance than the proposed PBM tracker for two types of errors and worse for the other two. However, we note that the KSP avails of a geometry calibration stage from the 3D scene to the image plane that is instead not required by the proposed tracker.

Table 4 Segmentation accuracy of the different trackers (ETISEO)

	PBM	MS	CCMSPF	KSP
x_c difference	2.1%	7.8%	15.8%	6.6%
y_c difference	7.7%	15.6%	24.5%	4.5%
Width difference	11.6%	66.0%	51.4%	12.2%
Height difference	11.1%	53.2%	48.6%	3.2%

5.4 Accuracy analysis at the frame level

The analysis in this section aims to provide an in-depth comparison of the tracking performance of the different trackers for some selected, challenging tracks within each of the videos. To evaluate the performance, the position and size of the tracked target in each frame are compared against the corresponding measurements in the ground truth.

PETS 2009 - The target object in this trajectory follows a downwards '∞'-shape path. The challenges in this trajectory are frequent occlusions, frequent changes in direction, similar color appearance of the target with the occluding person, and significant variations in the target's size in the image plane. A collection of frames in Fig. 7, left column, shows the target (the person with dark pants in the top frame) and the tracking results with PBM as the area with the five rectangular regions superimposed. These frames give evidence of the correct tracking of the selected target.

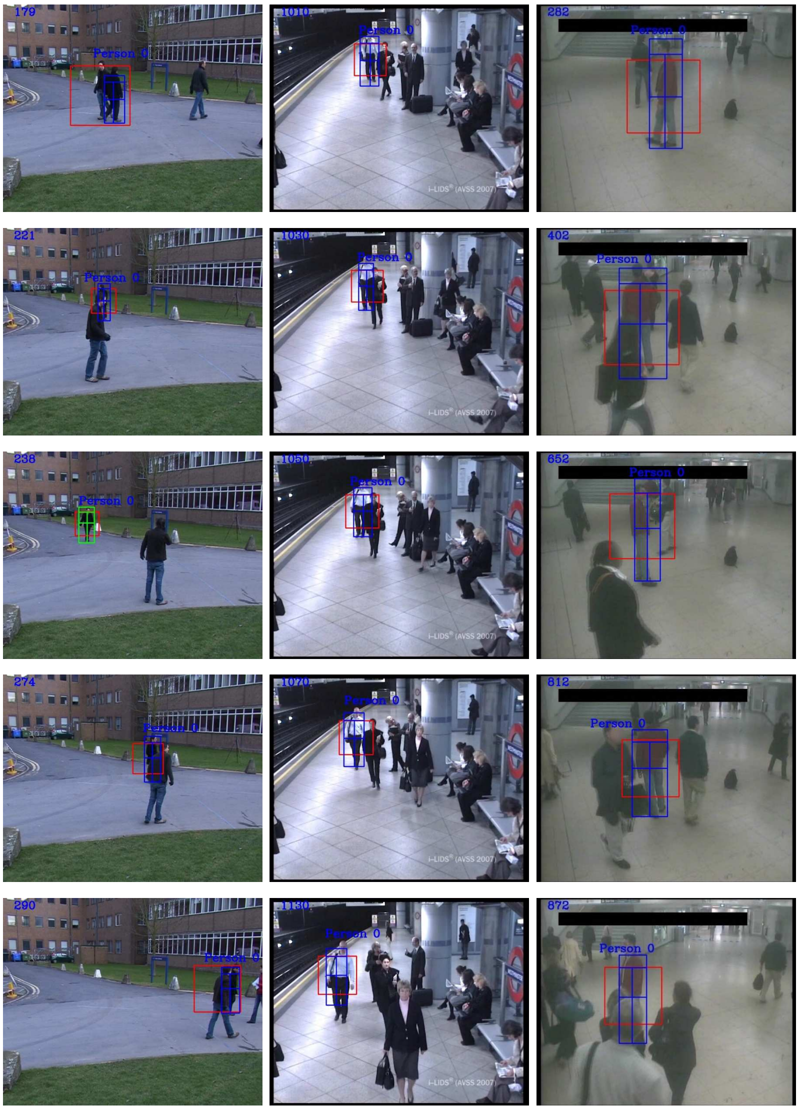


Fig. 7 Snapshots of selected trajectories and their corresponding tracking results from the three video sets.

Figure 8 shows the comparison of the extracted tracks with the ground truth in terms of target's centroid position. The plot shows that the PBM tracker is capable of accurately tracking the target throughout the trajectory while the other trackers report a major failure. The reason for PBM's performance is that it attempts data association at various bounding box's sizes and the layered approach proves capable of selecting a plausible size in most cases (for this sequence, its relative error on the estimate of the bounding box's width and height is within $\{-0.28, +0.38\}$ and $\{-0.12, +0.20\}$, respectively). Conversely, the MS tracker assigns multiple IDs to the target during the majority of the trajectory. The CCMSPF tracker's performance proves the worst in terms of both accuracy and continuity.

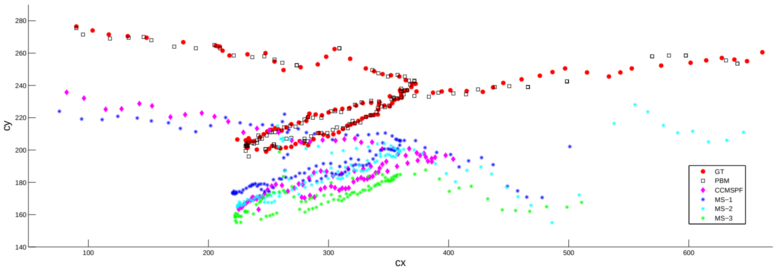


Fig. 8 PETS 2009: comparison of the target's centroid position for the various trackers and the ground truth. Horizontal axis: x-coordinate; vertical axis: y-coordinate; time is implicit. All tracks labelled as MS-n are multiple responses from the MS tracker. Although care was taken to make the dotted lines distinguishable, this figure and the next are better rendered in colours.

AVSS 2007 - Example frames for this trajectory are displayed in Fig. 7, mid column. The selected target (the man with bright shirt and bag) is partially occluded for a certain period of time as he walks toward the camera. Fig. 9 shows the comparison of the extracted tracks with the ground truth in terms of target's centroid position. The plot shows that the track from the PBM tracker is very close to the ground truth, while the track from CCMSPF is approximately 40 pixels distant on average, certainly as an artifact of the occlusion and the concentration of the particles' weight on the top part of the target. The track from MS is instead significantly delayed.

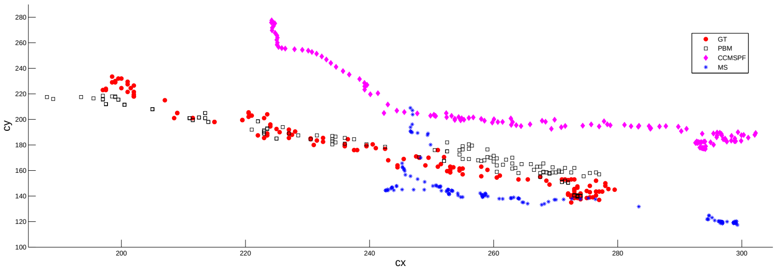


Fig. 9 AVSS 2007: comparison of the target's centroid position for the various trackers and the ground truth. Horizontal axis: x-coordinate; vertical axis: y-coordinate; time is implicit.

ETISEO - Example frames from a metro station are displayed in Fig. 7, rightmost column. In this trajectory, the target is circling around the center of the hall at a slow pace, but continuously changing in pose and direction. This sequence is made extremely challenging by the frequent and unpredictable occlusions between the target and the other subjects walking through the hall. In addition the video suffers from poor lighting conditions. Despite such issues, Fig. 7, right column, shows the successful tracking from PBM. Figure 10 shows the comparison of the extracted tracks with the ground truth in terms of target's centroid position. Only the PBM tracker proves capable of producing an accurate and unbroken trajectory throughout the sequence. MS, CCMSPF and also KSP produce, instead, several track fragments.

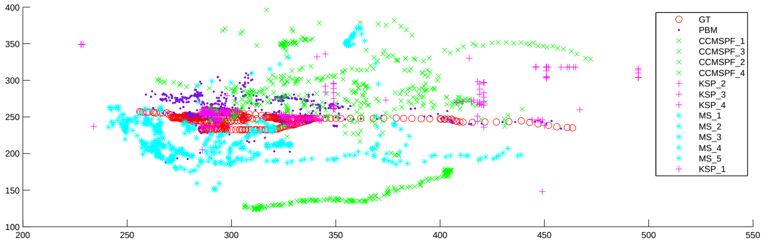


Fig. 10 ETISEO: comparison of the target's centroid position for the various trackers and the ground truth. Horizontal axis: x-coordinate; vertical axis: y-coordinate; time is implicit. All tracks labelled as MS-n, CCMSPF-n and KSP-n are multiple responses from the corresponding trackers.

For this dataset, we have also estimated the frame rate achievable by the proposed tracker: PBM can process 10 frames per second on an Intel Core i7 3.50 GHz PC. This performance is higher than that reported by the part-based tracker in [18] (5 fps), mainly because the frame size we use is smaller. Our tracker is not faster per se: rather, it can deliver a higher frame rate since it is based on a simpler model than [18] which can be fitted on frames of smaller resolution. In this sense, PBM proves an interesting approach for the low-medium resolution videos typical of current surveillance systems.

6 Conclusion

In this paper, we have proposed a novel tracker (Part-Based Model, PBM) based on a part-based model and layered data association, designed to track human subjects in scenes with frequent and heavy occlusions. The humam model is based on five parts which can be detected individually in order to withstand occlusions globally. In the proposed tracker, data association is provided by a layered approach, with correspondence hierarchically built between features (feature layer), parts (part layer) and, eventually, globally (global layer). The experimental results reported in the paper give evidence that this approach is effective in the designated scenario, with accuracy generally greater than that of the compared trackers. The experiments show that:

- the accuracy at the trajectory level (number of correct, missed, under-segmented, over-segmented trajectories, multiple IDs etc) has proven significantly higher for the proposed tracker (Table 1);
- the accuracy in locating the target's centroid has proven much higher for the proposed tracker than for of MS and CCMSPF, and comparable to that of KSP (Tables 2-4 and Figs. 8, 9, 10). A significant advantage of the proposed tracker over KSP is that it does not require any physical calibration stage;
- a similar comparative performance applies also for the target's estimated width and height (Tables 2-4).

The overall conclusion brought forward by this work is that a simplified part-based model offers a viable solution for video tracking of people in low-medium resolution video of public environments. As this scenario is common, the proposed solution can prove of benefit for a range of applications such as wide-area surveillance, media annotation, intelligent domotics and others.

References

1. AVSS2007 dataset: http://www.elec.qmul.ac.uk/staffinfo/andrea/avss2007 (2007)
2. Berclaz, J., Fleuret, F., Tretken, E., Fua, P.: Multiple object tracking using k-shortest paths optimization. IEEE Transactions on Pattern Analysis and Machine Intelligence **33**(9), 1806 – 1819 (2011)
3. Dempster, A., Laird, N., Rubin, D., et al.: Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society. Series B (Methodological) **39**(1), 1–38 (1977)
4. Felzenszwalb, P., Mcallester, D., Ramanan, D.: A Discriminatively Trained, Multiscale, Deformable Part Model. In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) 2008 (2008)
5. Garcia, C., Tziritas, G.: Face detection using quantized skin color regions merging and wavelet packet analysis. IEEE Transactions on Multimedia **1**(3), 264 –277 (1999)
6. KaewTraKulPong, P., Bowden, R.: An improved adaptive background mixture model for real-time tracking with shadow detection. In: Proc. European Workshop Advanced Video Based Surveillance Systems, vol. 1:3 (2001)
7. Kasturi, R., Goldgof, D., Soundararajan, P., Manohar, V., Garofolo, J., Bowers, R., Boonstra, M., Korzhova, V., Zhang, J.: Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol. IEEE Trans. Pattern Anal. Mach. Intell. **31**(2), 319–336 (2009)
8. Kittler, J., Hatef, M., Duin, R., Matas, J.: On combining classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence **20**(3), 226 –239 (1998)
9. Kjellstrom, H., Kragic and, D., Black, M.: Tracking people interacting with objects. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 747 –754 (2010)
10. Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2008)
11. Lin, Z., Davis, L.: Shape-based human detection and segmentation via hierarchical part-template matching. IEEE Transactions on Pattern Analysis and Machine Intelligence **32**(4), 604 –618 (2010)

12. Lin, Z., Davis, L., Doermann, D., DeMenthon, D.: Hierarchical part-template matching for human detection and segmentation. In: IEEE 11th International Conference on Computer Vision, ICCV 2007, pp. 1 –8 (2007)
13. Nghiem, A., Bremond, F., Thonnat, M., Valentin, V.: ETISEO: Performance evaluation for video surveillance systems. In: Proceedings of AVSS 2007 (2007)
14. Nummiaro, K., Koller-Meier, E., Van Gool, L.: An adaptive color-based particle filter. Image and Vision Computing **21**(1), 99–110 (2003)
15. PETS2009 dataset: http://www.cvg.rdg.ac.uk/pets2009/a.html (2009)
16. Ramesh, D., Meer, P.: Real-time tracking of non-rigid objects using mean shift. In: IEEE Conf. Computer Vision and Pattern Recognition (CVPR) 2000, vol. 2, pp. 142–149. Citeseer (2000)
17. Senior, A., Hampapur, A., Tian, Y., Brown, L., Pankanti, S., Bolle, R.: Appearance models for occlusion handling. Image and Vision Computing **24**(11), 1233–1243 (2006)
18. Shu, G., Dehghan, A., Oreifej, O., Hand, E., Shah, M.: Part-based multiple-person tracking with partial occlusion handling. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1815–1821 (2012)
19. Tran, S., Lin, Z., Harwood, D., Davis, L.: Umd_vdt, an integration of detection and tracking methods for multiple human tracking. In: R. Stiefelhagen, R. Bowers, J. Fiscus (eds.) Multimodal Technologies for Perception of Humans, *Lecture Notes in Computer Science*, vol. 4625, pp. 179–190. Springer Berlin / Heidelberg (2008)
20. Venkatesh, B.S., Descamps, A., Carincotte, C.: Counting people in the crowd using a generic head detector. In: Advanced Video and Signal-Based Surveillance 2012, pp. 470–475 (2012)
21. Wu, B., Nevatia, R.: Tracking of multiple, partially occluded humans based on static body part detection. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 951 – 958 (2006). DOI 10.1109/CVPR.2006.312
22. Yilmaz, A., Javed, O., Shah, M.: Object tracking: A survey. ACM Comput. Surv. **38** (2006)
23. Z. Zhang H. Gunes, M.P.: Head detection for video surveillance based on categorical hair and skin colour models. In: 16th IEEE Int. Conf. on Image Processing, pp. 1137 –1140 (2009)
24. Zhao, Q., Kang, J., Tao, H., Hua, W.: Part based human tracking in a multiple cues fusion framework. In: 18th International Conference on Pattern Recognition, ICPR 2006, vol. 1, pp. 450 –455 (2006). DOI 10.1109/ICPR.2006.914