Segmentation and tracking of multiple moving objects for intelligent video analysis

L-Q Xu, J L Landabaso† and B Lei

This paper aims to address two of the key research issues in computer vision — the detection and tracking of multiple objects in the cluttered dynamic scene — that underpin the intelligence aspects of advanced visual surveillance systems aiming at automated visual events detection and behaviour analysis. We discuss two major contributions in resolving these problems within a systematic framework. Firstly, for accurate object detection, an efficient and effective scheme is proposed to remove cast shadows/highlights with error corrections based on a conditional morphological reconstruction. Secondly, for effective tracking, a temporal-template-based tracking scheme is introduced, using multiple descriptive cues (velocity, shape, colour, etc) of the 2-D object appearance together with their respective variances over time. A scaled Euclidean distance is used as the matching metric, and the template is updated using Kalman filters when a matching is found or by linear mean prediction in the case of occlusion. Extensive experiments are carried out on video sequences from various real-world scenarios. The results show very promising tracking performance.

1. Introduction

In recent years, there has been considerable interest in visual surveillance of a wide range of indoor and outdoor sites by various parties. This is manifested by the widespread and unabated deployment of CCTV cameras in public and private areas. In particular, the increasing connectivity of broadband wired and wireless IP networks, and the emergence of IP-CCTV systems with smart sensors, enabling centralised or distributed remote monitoring, have further fuelled this trend. It is not uncommon nowadays to see a bank of displays in an organisation showing the activities of dozens of surveillance sites simultaneously. However, the limitations and deficiencies, together with the costs associated with human operators in monitoring the overwhelming video sources, have created urgent demands for automated video analysis solutions. Indeed, the ability of a system to automatically analyse and interpret visual scenes is of increasing importance to decision making, offering enormous business opportunities in the sector of information and communications technologies.

In monitoring a visual scene that is cluttered and busy, the importance of detection and tracking of any number of moving objects of interest can never be overestimated. This is the central element of an object-based intelligent video surveillance system, of which the two types of application are:

- to allow real-time detection of unforeseen events that warrant the attention of security guards or law enforcement officers to take preventive actions [1],
- to enable tagging and indexing of interesting (customer-defined) scene activities/statistics into a metadata database for rapid forensic analysis [2].

In addition, object detection and tracking are the building blocks of higher-level vision-based or assisted event monitoring and management systems with a view to understanding the complex actions, interactions, and abnormal behaviours of objects in the scene. The range of applications include detection of criminal behaviours in banks [3], marketing data analysis in shopping malls [4, 5], and well-being monitoring at home [6].

1.1 Surveillance systems — challenges

Vision-based surveillance systems can be classified in several different ways, depending on the conditions in which they are designed to operate:

- indoor, outdoor or airborne,

† Technical University of Catalunya, Spain

- the type and number of sensors,
- the objects and level of details to be tracked.

In this paper we focus on processing videos captured by a single fixed outdoor CCTV camera overlooking areas where there are a variety of vehicle and/or people activities, though the techniques developed can be applied to indoor scenarios.

There are typically a number of challenges associated with the chosen scenario in a realistic surveillance application environment.

- Natural cluttered background

  A natural outdoor environment is usually noisy and difficult to characterise. The video sequences captured are also often subjected to a compression process such as MPEG or JPEG before being transmitted via a network or stored for analysis. This introduces coding-induced noise into the already noisy imaging sources.

- Dynamic background

  The scene background is not normally a fixed structure, but often changes with time. In the case of a swaying tree or flag, each pixel in the background cannot be fully characterised by a single colour; two or more different appearances could be alternating.

- Illumination changes

  Outdoor surveillance systems suffer heavily from the change of weather conditions. Rain, sunset, sunrise, as well as floating clouds can have a dramatic impact on the scene illumination. Hence, they will degrade the performance of object detectors and trackers if these factors are not accommodated properly.

- Occlusions

  In a typical outdoor scene with many moving objects, occlusion is a crucial issue that needs special treatment. The occlusion can happen in the following cases:

  — inter-object where objects occlude each other — this problem becomes acute when two or more objects enter into the scene while occluding each other,

  — thin scene structures — thin objects in the scene such as trees or streetlights break a moving object into several (typically two) separate parts,

  — large scene structures — because of large scene structures such as buildings, moving objects may disappear completely for a period of time, and then re-appear, e.g. a pedestrian walking behind a parked or moving van,

- Object entries and exits

  Before a newly detected object in the scene is confirmed, it is important to know if this is a new entry, and if so, how it is to be modelled, and equally important is the decision about how and when to delete an existing object after its track is lost from the scene for some time,

- Shadows and highlights

  These are more problematic when tracking is carried out in outdoor environments, as very strong shadows or long shadows, larger than the actual object, are not uncommon; in addition, there are two types of shadow that need different treatment:

  — cast-shadows — these are areas in the background projected by an object in the direction of light rays, which can, without careful consideration, be easily taken as part of an object, causing difficulties to the ensuing object tracking and classification tasks,

  — self-shadows — these are parts of the object that are not illuminated by direct light, which a simple shadow-removal procedure is likely to get rid of, resulting in an inaccurate object silhouette.

1.2 Related work

These technical challenges, together with the ever-increasing demand of intelligent video surveillance applications, have led over recent years to extensive research activities that propose various new ideas, solutions and frameworks for robust object detection and tracking [7, 8]. Most adopt a type of 'background subtraction' technique to firstly detect foreground pixels. A connected component analysis (CCA) is then followed to cluster and label the foreground pixels into separate meaningful blobs, from which some inherent appearance and motion features can be extracted. Finally, there is the blob-based tracking process aiming to find persistent blob correspondences between consecutive frames. In addition, most application systems also deal with the issues of object categorisation or identification (and possibly detailed parts analysis) either before [7] or after [9] the tracking is established.

With regard to the 'background subtraction' technique, the background scene structures are usually modelled pixel-wise by various statistically based learning techniques using features such as intensities, colours, edges, textures, etc [10, 11]. The models employed can be a uni-modal Gaussian [12, 13], a

Gaussian mixture [14, 15], a non-parametric kernel density function [16], or simply temporal median filtering [9]. The issues of evaluation and maintenance of background models are discussed by Gao et al [17] and Toyama et al [18].

One major issue in background subtraction concerns shadow detection and removal [19]. An effective shadow removal scheme should remove completely the cast shadows, but not distort a foreground object's shape by removing extremities or deleting possible self-shadows. The use of a colour constancy model for shadow detection has been well studied by Horprasert et al [20], assuming that the chromaticity be the same while only intensity differs between the shadow and background. However, in the case where shadow removal based on colour properties alone may not be effective or colour information is not available, variants of gradient information can be exploited to fulfil the task [21]. Combinations of multiple cues (e.g. colour, normalised colour, gradient) were also considered by Javed et al [11] and McKenna et al [13]. Often, appropriate heuristic rules have to be adopted [21, 13] in order to accurately recover the true shape of an object.

Regarding the matching method and the choice of suitable metrics, the inherent heterogeneous nature of features extracted from the 2-D blobs has motivated some researchers to use only a few features, e.g. the size and velocity [8] for motion correspondence, and the size and position with Kalman predictors [14]. Others using more features conducted the matching in a hierarchical manner, e.g. in the order of centroid, shape, and then colour as discussed by Zhou and Aggarwal [9]. Note that if certain *a priori* factors are known, e.g. the type of an object to be tracked is a single person, then a more complex dynamic appearance model of the silhouette can be employed [7]. Also, in Elgamal et al [16], the kernel density function was used to model the colour distribution of an object to help detect and track individual persons who start to form a group and occlude each other; McKenna et al [13] provides another relevant example where probabilistic object models were exploited. Furthermore, domain knowledge of a physical site can be built beforehand for more effective management of object entry and exit [22] and for better handling the object occlusion issues in some applications [23].

In this paper we describe an effective multi-object detection and tracking system in which a few novel ideas are introduced to deal with these challenging issues. This leads to the enhancement of several aspects of state-of-the-art object detection and tracking techniques. In particular, we employ a technique to suppress the falsely detected foreground pixels, caused mainly by video compression artefacts. A novel framework is introduced for effective cast shadows/highlights removal while preserving the original object shape. An integrated matching strategy is discussed, using the scaled Euclidean distance metric, in which a feature set characterising a foreground object is used simultaneously, taking care of both the scale and variance of each of the features. This matching method is not only robust (in the sense of tolerating sudden speed change or direction change), but also allows an easy inclusion of more extracted features, if necessary, leaving room for future enhancement. Figure 1 depicts schematically the block diagram of our proposed object detection and tracking system, which comprises two named major functional modules, each in turn containing a number of processing steps. The object classification module is included for completeness, though it will not be discussed in this paper; interested readers are referred to Javed and Shah [8] or Zhou and Aggarwal [9] for more information.

The paper is structured as follows. In the next section, techniques for pixel-domain analysis, leading to segmented foreground object blobs, are discussed, with emphasis on the introduction of a novel shadow removal scheme. Section 3 is devoted to discussion of multi-object tracking, including the use of a temporal object template, the adoption of a parallel matching procedure and the partial occlusion handling. Section 4 presents the experimental studies of this system with various real-world test sequences undergoing a variety of video compression procedures. The paper concludes in section 5 with a discussion of future research direction and system enhancement.

2. Moving objects segmentation with shadow removal

As depicted in Fig 1, the first issue to be addressed is 'background learning', designed for segmenting scene pixels forming part of the foreground moving objects via background subtraction. As in Javed and Shah [8], the adaptive background learning method proposed by Stauffer and Grimson [14] is adopted. At each pixel location, a Gaussian mixture model (GMM) is used to model the temporal colour variations in the imaging scene. The Gaussian distributions are updated with each incoming frame; the models are then used to determine if an incoming pixel is generated by the background process or a foreground moving object. This model allows a proper representation of the background scene undergoing slow and smooth lighting changes (but not suddenly turning on or off, e.g. caused by floating clouds) and momentary and random variations such as trees or flags swaying in the wind.

Considering that the foreground pixels thus obtained are likely to suffer from false detections due to imaging and compression noise as well as camera jitter,

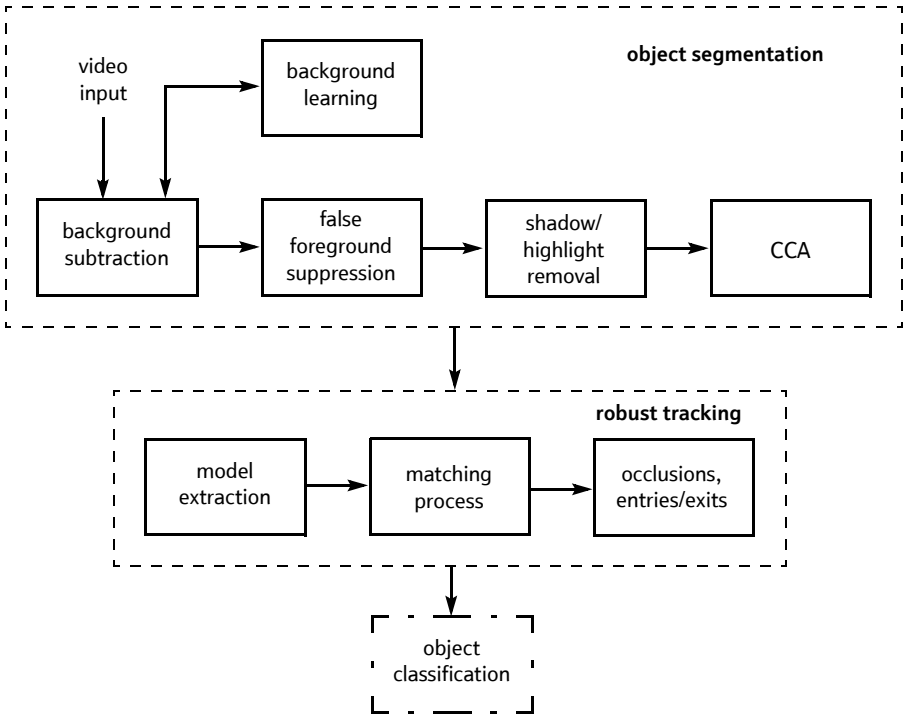

Fig 1 The schematic system block diagram showing the two main functional modules.

a false-foreground-pixel-suppression procedure is introduced to alleviate this problem. The idea is that, for each pixel $\boldsymbol{x} = \{x, y\}$ initially classified as a foreground pixel, the GMMs of its 8-connected neighbouring pixels are examined. If the majority of them (>5) agree that $\boldsymbol{x}$ is a background pixel, then $\boldsymbol{x}$ is considered as a false detection and removed from foreground.

2.1 A novel shadow/highlight removal scheme

Once the foreground pixels are identified, a further detection scheme is applied to locate areas likely to be cast shadows or highlights. In the following, we discuss a novel scheme for effective shadow (highlights) detection using both colour and texture cues. Since in any shadow-removal algorithm, misclassification errors often occur, resulting in distorted object shapes, the core of this scheme is the use of a technique capable of correcting these errors. The technique is based on a greedy thresholding followed by a conditional morphological dilation. The greedy thresholding removes all shadows together with some true foreground pixels. The conditional morphological dilation then aims to recover only those deleted true foreground pixels constrained within the original foreground mask.

The working mechanism of this novel scheme is shown in Fig 2 and comprises the following four steps.

- Colour-based detection

  As the first step, a simplified version of the colour constancy model introduced by Horprasert et al [20] is employed. This model evaluates the

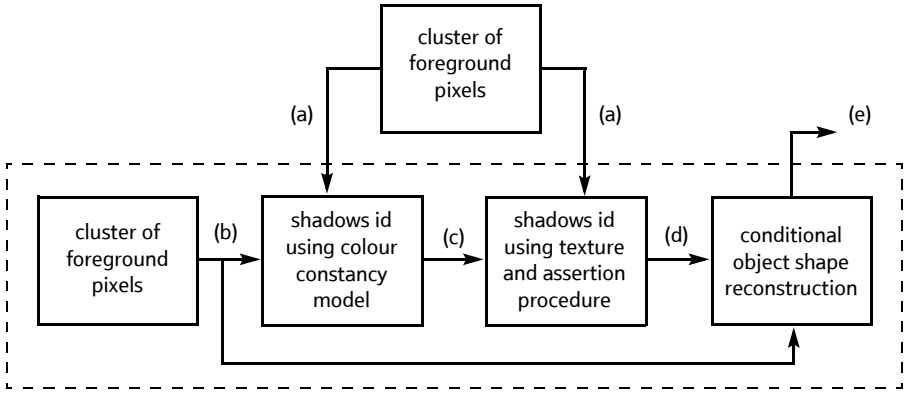

Fig 2 The schematic diagram of the novel shadows/highlights removal approach made up of four main processing steps. The input and output of each block are as follows — (a) the adaptive background image; (b) initial foreground segmentation result; (c) shadows/highlights removal using colour constancy model; (d) the result after shadows assertion using gradient/texture information, generating a 'skeleton' image; and (e) final reconstructed foreground regions.

variability in both brightness and colour distortions in RGB colour space between the foreground pixels and the adaptive background. The background reference image is obtained from the mean of the most probable Gaussian component of the GMM modelling each pixel. Possible shadows and highlights are then detected by certain thresholding decisions. It was observed though that this procedure is less effective in cases where the objects of interest have similar colours to those of presumed shadows.

- Texture-based detection

  The same regions with or without cast shadows tend to retain similar texture (edge) properties despite the difference in illumination. To exploit this fact, the Sobel edge detector is used to compute the horizontal and vertical gradient for both the foreground pixels and their corresponding background ones. For each pixel, the Euclidean distance with respect to the gradients is evaluated over a small neighbourhood region, which is then employed to examine the similarity between the foreground and reference pixel. If the distance is less than a certain threshold, then a possible shadow pixel is suggested.

- Assertion procedure

  Based on the detection results from the above two steps, an assertion procedure is introduced, which confirms a pixel as belong to foreground only if both the above two outputs agree. Output from this procedure is a seed 'skeleton' image (as shown in Fig 4(c)) free of shadows and highlights.

- Conditional object shape reconstruction

  The above processing steps are designed to effectively remove cast shadows and highlights, though they also invariably delete some foreground object pixels (self-shadows), causing the distortion of a real object's shape. Therefore, a morphology-based conditional region reconstruction step is employed to restore each object's original shape from the 'skeleton' image.

The mathematical morphology reconstruction filter uses an image called 'marker' as the seed to rebuild an object inside an original image called 'mask'. In our case, the 'marker' image (see Fig 4(c)) is a binary image in which a pixel is set at '1' when it corresponds to a foreground, not a cast shadow/highlight, pixel. On the other hand, the 'mask' image (see Fig 4(b)) is also a binary image where a '1' pixel can correspond to a foreground pixel, or a cast shadow/highlight pixel, or speckle noise.

It is highly desirable that the 'marker' image, $\tilde{M}$ contains only real foreground object pixels, i.e. not any shadow/highlight pixels so that those regions will not be reconstructed. Therefore, the use of very aggressive thresholds is necessary in the colour-based removal process to ensure that all the shadow/highlight pixels are removed. A speckle noise removal filter is also applied to suppress isolated noisy foreground pixels that remain and to obtain a good quality 'marker' image, $\tilde{M}$.

The speckle removal filter is also realised using morphological operators as shown in equation (1):

$$\tilde{M} = M \cap (M \oplus N) \qquad \ldots\ldots (1)$$

where M is the binary image generated after shadow removal and assertion process; N denotes the structuring element, shown in Fig 3, with its origin at the centre.

The dilation operation $M \oplus N$ in equation (1) identifies all the pixels that are four-connected to (i.e. next to) a pixel of M. Hence, $\tilde{M}$ identifies all the pixels that are in M and also have a four-connected neighbour, eliminating the isolated pixels in M.

As a result, only the regions not affected by noise which are clearly free of shadows/highlights (Fig 4(c)) are subject to the shape reconstruction process shown in equation (2):

$$R = M_s \cap (\tilde{M} \oplus SE) \qquad \ldots\ldots (2)$$

where M_s is the 'mask', $\tilde{M}$ the 'marker', and SE the structuring element whose size usually depends on the size of the objects of interest, although a 9×9 square element proved to work well in our tests.

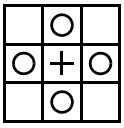

Fig 3 The 3×3 morphological structuring element used for speckles filtering. Note that the origin is not included.

Basically this process consists of a dilation of the 'marker' image, followed by the intersection with the 'mask' image. The underlying idea is that there should be a fairly large number of valid object pixels remaining after the shadow removal processing. These pixels are appropriate for leading the reconstruction of neighbouring points as long as they form part of the silhouette in the original blob (prior to the shadow removal as in Fig 4(b)). The finally reconstructed blobs are shown in Fig 4(d).

This novel combined scheme gives favourable results compared to the current state-of-the-art ones to

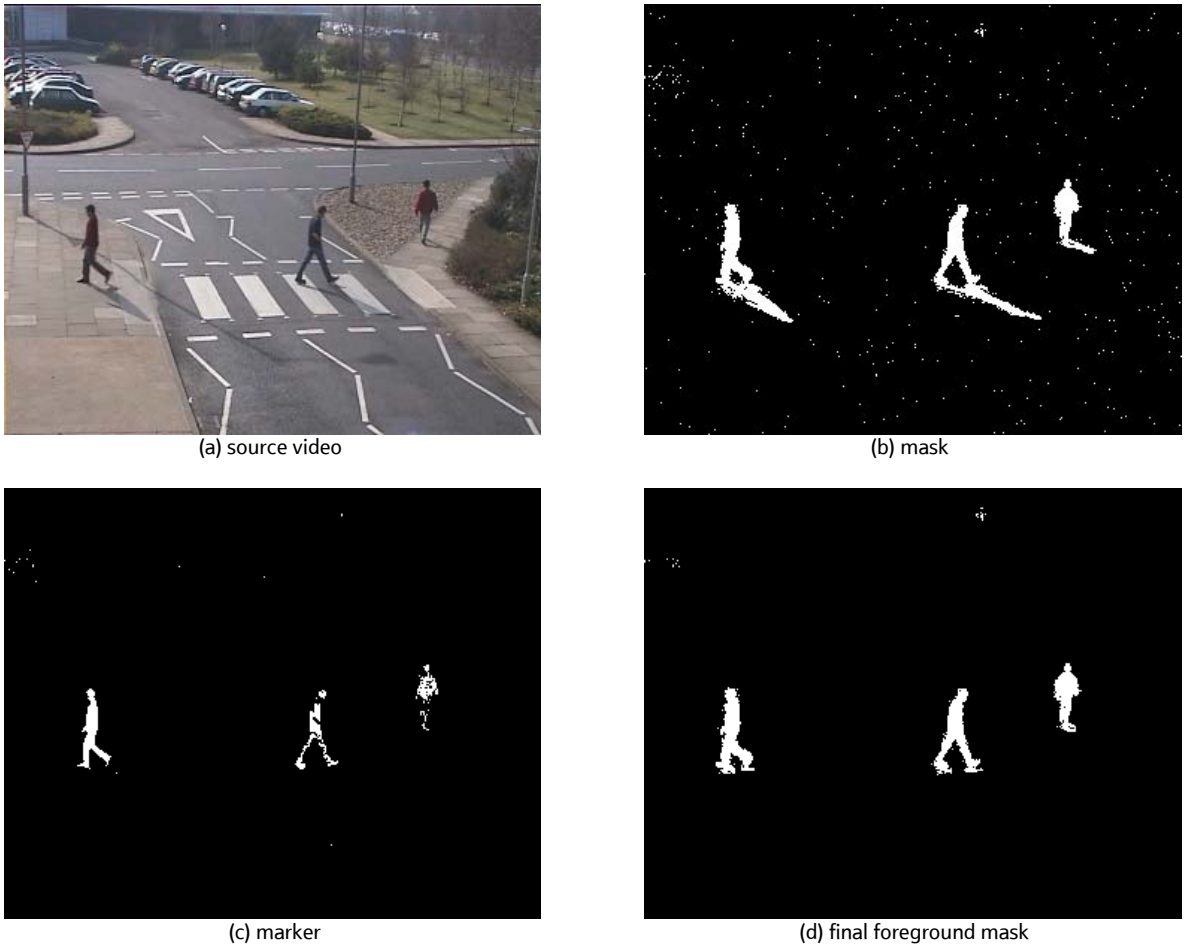

Fig 4 (a) A snapshot of a surveillance video sequence, the cast shadows from pedestrians are strong and large; (b) the result of initial foreground pixels segmentation, the moving shadows being included; (c) the 'skeleton' image obtained after the shadow removal processing; and (d) the final reconstructed objects with erroneous pixels corrected.

suppress shadows/highlights. Figure 4 illustrates an example of this scheme at various processing stages.

3. Multi-object tracking using temporal templates

After the cast shadows/highlights removal procedure, a classic 8-connectivity connected component analysis (CCA) is performed to group all the pixels presumably belonging to individual objects into respective blobs. The blobs are temporally tracked throughout their movements within the scene by means of temporal templates. Figure 5 illustrates an example where the three objects (indexed by l) are tracked to frame t, which seek to match the newly detected candidate blobs (indexed by k) in frame $t+1$. One of these four candidates (near the right border) just enters into the scene, for which a new template has to be created.

3.1 Temporal templates

Each object of interest in the scene is modelled by a temporal template of persistent characteristic features. In the current studies, a set of five significant features is

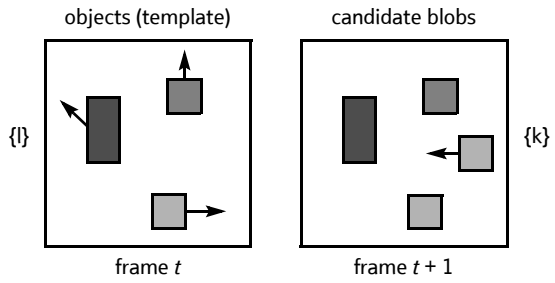

Fig 5 The illustration of object tracking between two consecutive frames. On the left are the three objects already tracked, for which feature template models exist; on the right are the four newly detected candidate blobs in frame $t+1$, for which matching to the corresponding tracks are sought, noting the far right one just enters the viewing scene.

used, describing the velocity, shape, and colour of each object (candidate blob) as shown in Table 1.

Therefore at time t, we have, for each object l centred at (p_{lx}, p_{ly}), a template of features:

$$M_l(t) = (\mathbf{v}_l, s_l, r_l, \theta_l, \mathbf{c}_l)$$

There are two points that need special clarification as follows:

Table 1 The five significant features for each object.

$\boldsymbol{v} = (v_x, v_y)$	the velocity at its centroid (p_x, p_y)
s	the size, or number of pixels contained
r	the ratio of the major and minor axis of the best-fit ellipse of the blob [24]; it is a better descriptor of an object's posture than its bounding box
θ	the orientation of the major-axis of the ellipse
$\boldsymbol{c}$	the dominant colour, computed as the principal eigenvector of the colour co-variance matrix for pixels within the blob [9]

- prior to matching the template l with a candidate blob k in frame t+1, centred at (p'_{kx}, p'_{ky}) with a feature vector $B_k(t+1) = (\boldsymbol{v}'_k, s'_k, r'_k, \theta'_k, \boldsymbol{c}'_k)$, Kalman filters are used to update the template by predicting, respectively, its new velocity, size, aspect ratio, and orientation in $\hat{M}_l(t+1)$ — the velocity of the candidate blob k is calculated as:

$$\boldsymbol{v}'_k = (p'_{kx}, p'_{ky})^T - (p_{lx}, p_{ly})^T$$

- the difference between the dominant colour of template l and that of candidate blob k is defined in equation (3):

$$d_{lk}(\boldsymbol{c}_l, \boldsymbol{c}'_k) = 1 - \frac{\boldsymbol{c}_l \bullet \boldsymbol{c}_k}{\|\boldsymbol{c}_l\| \cdot \|\boldsymbol{c}_k\|} \qquad \ldots\ldots (3)$$

The mean $\bar{M}_l(t)$ and variance $V_l(t)$ vector of a template l are updated when a matching candidate blob k is found. And they are computed using the most recent L blobs on the track, or over a temporal window of L frames (e.g. $L = 50$). The set of Kalman filters, $KF_l(t)$, is updated by feeding with the corresponding feature value of the matched blob.

It is clear that the variance of each template feature should be analysed and taken into account in the matching process outlined in section 3.2 to achieve a robust tracking result.

3.2 Matching procedure

We choose to use a parallel matching strategy in preference to the serial matching ones such as that used by Zhou and Aggarwal [9]. The next issue is to employ a proper distance metric that best suits the problem under study. As described above, the template for each object being tracked has a set of associated Kalman filters, each of which predicts the expected value for one feature (except for the dominant colour) in the next frame. Obviously, some features are more persistent for an object, while others may be more susceptible to noise, and different features normally assume numerical values of different scales and variances. Euclidean distance does not account for these factors as it will allow dimensions with larger scales to dominate the distance measure.

One way to tackle this problem is to use the Mahalanobis distance metric, which takes account of not only the scale and variance of a feature, but also its correlation with other features based on the co-variance matrix. Thus, if there are correlated features, their contributions are weighted appropriately.

Though, for simplicity, in the current work, a scaled Euclidean distance shown in equation (4) is adopted to match the template l and a candidate blob k, assuming a diagonal co-variance matrix. For a heterogeneous data set, this is a reasonable distance definition:

$$D(l, k) = \sqrt{\sum_{i=1}^{N} \frac{(x_{li} - y_{ki})^2}{\sigma_{li}^2}} \qquad \ldots\ldots (4)$$

where the index i runs through all the N=5 features of the template, and σ_{li}^2 is the corresponding component of the variance vector $V_l(t)$. Note exceptionally that, as discussed in section 3.1, on the dominant colour feature, it can be viewed as, $x_{li} - y_{ki} = d_{lk}(\boldsymbol{c}_l, \boldsymbol{c}'_k)$. The initial values of all components of $V_l(t)$ are either set at a relatively large value or inherited from a neighbouring object.

Having defined a suitable distance metric, the matching process can be described in greater detail as follows.

Given that in frame t, for each object l being tracked so far, we have:

$M_l(t)$	the template of features
$(\bar{M}_l(t), V_l(t))$	its mean and variance vectors
$KF_l(t)$	the related set of Kalman filters
$TK(t) = n$	the counter of tracked frames, i.e. current track length
$MS(t) = 0$	the counter of lost frames
$\hat{M}_l(t+1)$	the expected values in frame t+1 by Kalman prediction

- Step 1

  For each new frame t+1, all the valid candidate blobs $\{k\}$ are matched against all the existing tracks $\{l\}$ via equation (4) by way of the template prediction, $\hat{M}_l(t+1)$, variance vector $V_l(t)$ and $B_k(t+1)$. A ranking list is then built for each object l by sorting the matching pairs from low to high cost. The matching pair with the lowest cost value $D(l, k)$, which is also less than a threshold, THR (e.g.10 in our experiments), is identified as a match pair.

- Step 2

  If object l is matched by a candidate blob k in frame $t+1$, then the track length $TK(t+1)$ is increased by 1, and the normal updates for l are performed. We

obtain $M_l(t+1) = B_k(t+1)$, as well as the mean and variance ($\bar{M}_l(t+1)$, $V_l(t+1)$) respectively, as discussed in section 3.1, and correspondingly the Kalman filters $KF_l(t+1)$.

- Step 3

  If object l has found no match at all in frame $t+1$, presumably because it is missing or occluded, then the mean of its template is kept the same, or $\bar{M}_l(t+1) = \bar{M}_l(t)$; the lost counter $MS(t+1)$ is increased by 1. The object l is carried over to the next frame, though the following rules apply:

  — if object l has been lost for a certain number of frames, or $MS(t+1) \geq MAX\_LOST$ (e.g. 10), then it is deleted from the scene; the possible explanations include becoming static (merged into background), entering into a building/car, or leaving the camera's field of view,

  — otherwise, the variance $(V_l(t+1)$ is adjusted according to equation (5) to assist the tracker to recover the lost object that may undergo unexpected or sudden movements:

  $$\sigma_i^2(t+1) = (1+\delta)\sigma_i^2(t) \qquad \ldots\ldots (5)$$

  where $\delta = 0.05$ is a good choice. As no observation is available for each feature, the latest template mean vector is used for prediction, which states that $M_l(t+1) = M_l(t) + \bar{M}_l(t)$.

  Note that the MAX_LOST is measured in terms of number of frames; in actual applications the value should be adjusted in accordance with the video capture frame rate and maximum speed of a moving object, if possible.

- Step 4

  For each candidate blob k in frame $t+1$ that is not matched, a new object template $M_k(t+1)$ is created from $B_k(t+1)$. The choice of initial variance vector $V_k(t+1)$ needs some consideration — it can be copied from either a very similar object already in the scene or typical values obtained by prior statistical analysis of tracked objects. This new object, however, will not be declared (marked) until after it has been tracked for a number of frames, or $TK(t+1) >= MIN\_SEEN$ (e.g. 10), so as to discount any short momentary object movements; otherwise it will be deleted.

3.3 Occlusions handling

In the current approach, no use is made of any special heuristics on areas where an object may enter (exit) into (from) the scene. The possible background structures that may occlude moving foreground objects are also unknown *a priori* [23]. Objects may just appear or disappear in the middle of the image, and, hence, positional rules are not enforced, as opposed to Stauffer [22].

To handle the occlusion issue with *a priori* information, a simple heuristic is adopted. Every time an object fails to find a matching candidate blob (step 3, section 3.2), a test on occlusion is carried out. If the object's predicted bounding box overlaps a certain new candidate's bounding box, then this object is marked as 'occluded'. If this new candidate occludes more than one 'unmatched' object, it is deleted. The template of each 'occluded' object is blindly updated as discussed above from the previous tracking results until it gets matched again or removed after being missing for certain frames.

As discussed before, during the possible occlusion period, the object template of features is updated using the average of the last 50 correct predictions to obtain a long-term tendency prediction. Occluded objects are better tracked using the averaged template predictions. In doing so, small erratic movements in the last few frames are filtered out. Predictions of positions are constrained within the blob that occludes the current 'occluded' object.

4. Experimental results

The system has been evaluated extensively on standard test sequences such as the set of benchmarking image sequences provided by PETS'2001 and a range of our own captured image sequences under various weather conditions and video compression formats.

For PETS sequences, original images are provided in JPEG format, and their frame size is 768×576 pixels. In our experiments though, the sub-sampled images of size 384×288 pixels were used. Also, an AVI video file was created for each image sequence using an XviD codec, introducing a second temporal compression. Apart from these compression artefacts, the imaging scenes also contain a range of difficult defects, including thin structures, window reflections, illumination changes due to slowly moving clouds, and swaying leaves in trees. Our system has dealt with all these problems successfully, and handles very well the complex occlusion situations. Figure 6 shows an example where the white van is occluded by a thin structure, or streetlight pole (left), and subsequently a group of people are largely blocked by the van for a few frames (middle).

For the other sequences, a CIF-size image frame (352×288 pixels) is used. The original video was captured at 25 fps using Mini DV format, and then converted to MPEG-1, followed by an XviD compression. Figure 7 illustrates an example of a complex and difficult situation where large and strong shadows

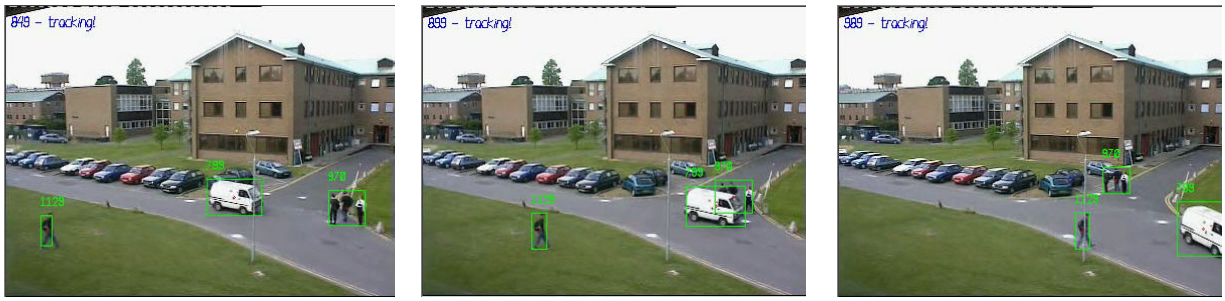

Fig 6 An example (from PETS'2001) illustrating one of the difficult tracking situations that the system handles successfully, in which the moving white van, first occluded by the thin streetlight pole, then partially occludes a group of walking people (from left to right): before, during, and after occlusion. The tracking labels have been correctly kept.

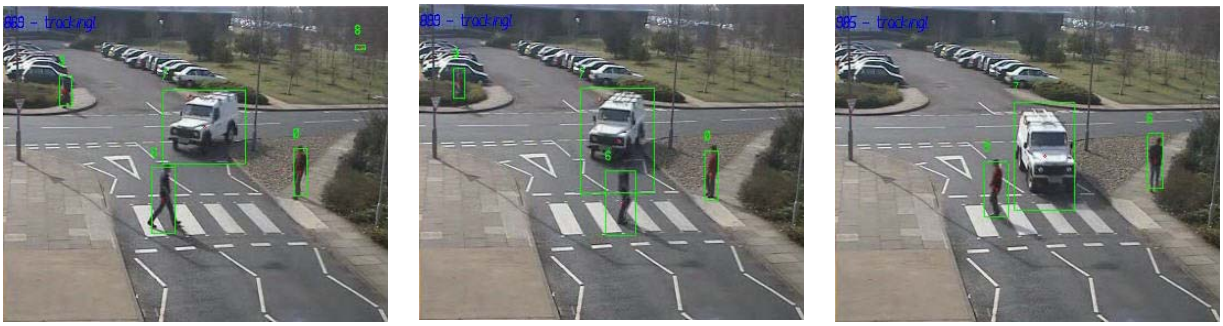

Fig 7 Another example illustrating the success of the system in dealing with severe shadows problem and complex dynamic occlusion situation. Two people were walking towards each other across the pedestrian crossing, whilst a van is approaching and slowing down (from left to right) — before, during, and after their intersection.

exist and three objects (two people and a van) pass by each other. Figure 8 gives another example displaying the results obtained after different processing stages of the system. The system runs at an average rate of 12 fps on a PC with a single 2 GHz Pentium-4 processor.

Fig 8 Results showing different processing stages of the system (anti-clockwise from top left) — the source video image overlaid with objects being tracked; the learned background image; the foreground mask output from initial background subtraction and thresholding; the final restored foreground mask after noise suppression and shadow removal.

Some problems occurred when a few individually moving objects start to join each other and form a group. These objects are correctly tracked within the limit of predefined *MAX_LOST* frames as if they were occluding each other. Beyond this limit the system decides that they have disappeared, and creates a new template for the whole group. Other problems may occur when objects abruptly change their motion trajectories during occlusions — sometimes the system is able to recover the individual objects after the occlusion, but on other occasions new templates are created.

The system copes with shadows and highlights satisfactorily in most cases, though very long cast shadows may not always be completely removed. A small defect of the algorithm is that the reconstructed region contains a small patch of shadow in an object's exterior where the cast shadow starts (see the feet of the people in Fig 4(d)). This patch is about half the size of the structuring element used, and is produced during the conditional dilation. Intersection with the mask image cannot suppress this segment as all the shadowed regions form part of the mask.

5. Conclusions

In this paper, we have presented a vision-based system for accurate segmentation and tracking of moving objects in cluttered and dynamic outdoor environments

surveyed by a single fixed camera. Each foreground object of interest has been segmented and shadows/highlights removed by an effective scheme. The 2-D appearance of each detected object blob is described by multiple characteristic cues including velocity, size, elliptic-fit aspect ratio, orientation, and dominant colour. This template of features is used, by way of a scaled Euclidean distance-matching metric, for tracking between object templates and the candidate blobs appearing in the new frame. In completing the system, we have also introduced technical solutions dealing with false foreground pixel suppression, and temporal template adaptation. Experiments have been conducted on a variety of real-world wide-area scenarios under different weather conditions. Good and consistent performance has been confirmed. The method has successfully coped with illumination changes, partial occlusions, clutters, and scale and orientation variations of objects of interest — and, especially, it is not sensitive to noise incurred by the camera imaging system and different video codec.

Having undertaken this first but significant step towards developing a fully functional intelligent video surveillance system, several aspects will be further explored to enhance the robustness and consistency.

- Shadow removal

  As previously noted, removing cast shadows while preserving self shadows is always a conflicting goal. Thanks to the skeleton-based conditional reconstruction method for error correction, we can start with a very greedy and simple shadow removal scheme. It works well most of the time, though in certain cases where a foreground object happens to have similar properties to that of the shadowed background, it would fragment the object into several smaller parts, thus causing problems for the tracking procedure. It is necessary to devise a new procedure to link those parts into a single object.

- Matching

  For the matching problem, currently all features involved are treated separately and identically. A further investigation could be done to evaluate the impact of each feature on the matching score, and then choose to use the more significant ones as well as determine their relative contributions in the final distance metric calculation.

- Occlusion

  As regards handling the occlusion problem, we have used a simple heuristic at the moment. It fails in dealing with more sophisticated multiple object occlusion or long total occlusions. The method can be improved if, during an object's presence in the scene, more tracking states than the current three ('matched', 'occluded', and 'disappeared') are introduced, plus employing more heuristic rules in the management of these state transitions. On the other hand, the use of a probabilistic texture [5] or colour appearance model [25] may help find a better solution to resolving occlusions, especially for people tracking indoor environments where more information is available concerning target objects and their interactions.

References

1 Lipton A J, Heartwell C H, Haering N and Madden D: 'Automated video protection, monitoring and detection of critical infrastructure', IEEE Aerospace and Electronic Systems Magazine, 18, No 5 (May 2003).

2 Perrott A J, Lindsay A T and Parkes A P: 'Realtime multimedia tagging and content-based retrieval for CCTV surveillance system', Proc of SPIE: Internet Multimedia Management Systems III, Boston (2002).

3 Georis B, Maziere M, Bremond F and Thonnat M: 'A video interpretation platform applied to bank agency monitoring', Proc of IEE IDSS'04, pp 46—50, London (February 2004).

4 Haritaoglu I and Flickner M: 'Detection and tracking of shopping groups in stores', Proc of IEEE CVPR'2001, Kauai, Hawaii, USA (December 2001).

5 Senior A: 'Tracking people with probabilistic appearance models', Proc 3rd IEEE Intl Workshop on Performance Evaluation of Tracking and Surveillance (PETS'2002), pp 48—55, Copenhagen, Denmark (June 2002).

6 Cucchiara R, Grana C, Prati A, Tardini G and Vezzani R: 'Using computer vision techniques for dangerous situation detection in domotics applications', Proc of IEE IDSS'04, pp 1—5, London (February 2004).

7 Haritaoglu I, Harwood D and Davis L: 'W4: real time surveillance of people and their activities', IEEE Trans. on Pattern Analysis and Machine Intelligence, 22, No 8 (August 2000).

8 Javed O and Shah M: 'Tracking and object classification for automated surveillance', Proc of ECCV'2002, Copenhagen, Denmark, pp 343—357 (May—June 2002).

9 Zhou Q and Aggarwal J K: 'Tracking and classifying moving objects from video', Proc of 2nd IEEE Intl Workshop on Performance Evaluation of Tracking and Surveillance (PETS'2001), Kauai, Hawaii, USA (December 2001).

10 Li L and Leung M K H: 'Integrating intensity and texture differences for robust change detection', IEEE Trans on Image Processing, 11, No 2, pp 105—112 (2002).

11 Javed O, Shafique K and Shah M: 'A hierarchical approach to robust background subtraction using color and gradient information', Proc of IEEE Workshop on Motion and Video Computing, Orlando, USA (December 2002).

12 Jabri S, Duric Z, Wechsler H and Rosenfeld A: 'Detection and location of people in video images using adaptive fusion of color and edge information', Proc of ICPR'2000, Barcelona, Spain (September 2000).

13 McKenna S J, Jabri S, Duric Z, Rosenfeld A and Wechsler H: 'Tracking groups of people', Computer Vision and Image Understanding, 80, pp 42—56 (2000).

14 Stauffer C and Grimson W E L: 'Learning patterns of activity using real-time tracking', IEEE Trans on Pattern Analysis and Machine Intelligence, 22, No 8 (August 2000).

15 Lee D S, Hull J J and Erol B: 'A Bayesian framework for Gaussian mixture background modelling', Proc of IEEE ICIP'2003, Barcelona, Spain (September 2003).

16 Elgamal A, Duraiswami R, Harwood D and Davis L: 'Background and foreground modelling using nonparametric kernel density estimation for visual surveillance', Proc of the IEEE, 90, No 7 (July 2002).

17 Gao X, Boult T E, Coetzee F and Ramesh V: 'Error analysis of background adaptation', Proc of IEEE CVPR'2000, South Carolina, USA, pp 503—510 (June 2000).

18 Toyama K, Krumm J, Brumitt B and Meyers B: 'Wallflower: principles and practice of background maintenance', Proc of IEEE ICCV'99, pp 255—261, Kerkyra, Greece (September 1999).

19 Cucchiara R, Grana C, Piccardi M and Prati A: 'Detecting moving objects, ghosts and shadows in video streams', IEEE Trans on Pattern Analysis and Machine Intelligence, 25, No 10, pp 1337—42 (2003).

20 Horprasert T, Harwood D and Davis L: 'A statistical approach for real-time robust background subtraction and shadow detection', Proc of ICCV'99 FRAME-RATE Workshop (1999).

21 Bevilacqua A: 'Effective shadow detection in traffic monitoring applications', Proc of WSCG'2003, Plzen-Bory, Czech Republic (February 2003).

22 Stauffer C: 'Estimating tracking sources and sinks', Proc of 2nd IEEE Workshop on Event Mining (in conjunction with CVPR'2003), 4, Madison, Wisconsin (June 2003).

23 Xu M and Ellis T J: 'Partial observation vs. blind tracking through occlusion', in Proc of BMVC'2002, Cardiff, pp 777—786 (September 2002).

24 Fitzgibbon A W and Fisher R B: 'A buyer's guide to conic fitting', Proc of 5th British Machine Vision Conference, Birmingham, pp 513—522 (1995).

25 Balcells Capellades M, Doermann D, DeMenthon D and Chellappa R: 'An appearance based approach for human and object tracking', Proc of IEEE ICIP'2003, Barcelona, Spain (September 2003).

Li-Qun Xu joined BT Research and Venturing in 1996 as a Senior Researcher, where he is currently a Principal Researcher and Project Manager in the Broadband Applications Research Centre. His recent research interests are in the broad areas of visual information processing, including multimedia content analysis and indexing, robust object segmentation and tracking for intelligent visual surveillance, people behaviour and event analysis, 2-D motion analysis and segmentation, 3-D vision techniques and image-based rendering for collaborative working environment, among others. He has published prolifically on these and allied topics and holds a number of patents and pending applications. Prior to his career with BT, he has worked as an academic in a number of British Universities, both as a member of the research staff and lately within the faculty between 1990 and 1996. He earned his PhD in Information Engineering from Southeast University, Nanjing, China, in 1988. He is a member of British Computer Society and a member of IEEE Signal Processing and Computer Societies.

Jose-Luis Landabaso is currently a PhD student in the Department of Signal Theory and Communications, Technical University of Catalunya (UPC), Spain. His thesis direction is in the area of dynamic visual scene understanding using a multi-camera system. He earned his MEng degree from UPC in June 2001. He then worked as a student intern in Philips Research, New York between June 2001 and February 2002, and in BT Research and Venturing at Adastral Park between October 2002 and April 2003. He has had several publications related to facial expression recognition based on MPEG-4 coding parameters, hidden Markov models and object segmentation and tracking.

Bangjun Lei received his BSc and MSc degree in Computer Science from Xiían Jiaotong University, China in 1995 and 1998, respectively. He then moved to Technical University, Delft, the Netherlands, in 1999, to pursue his PhD study, where he earned his PhD degree with a thesis entitled 'A viewpoint adaptive system for 3-D telepresence' in September 2003. He joined BT Research and Venturing as a Researcher in October 2003. His current research interest includes advanced low-level image processing techniques, 3-D imaging and image-based rendering, and computer vision for intelligent visual surveillance applications. He is a member of IEEE.