

# A real-time object detecting and tracking system for outdoor night surveillance<sup>☆</sup>

Kaiqi Huang<sup>a,\*</sup>, Liangsheng Wang<sup>a</sup>, Tieniu Tan<sup>a</sup>, Steve Maybank<sup>b</sup>

<sup>a</sup>National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100080, China

<sup>b</sup>School of Computer Science and Information Systems, Birkbeck College, Malet Street, London WC1E 7HX, UK

Received 27 April 2006; received in revised form 31 March 2007; accepted 23 May 2007

## Abstract

Autonomous video surveillance and monitoring has a rich history. Many deployed systems are able to reliably track human motion in indoor and controlled outdoor environments. However, object detection and tracking at night remain very important problems for visual surveillance. The objects are often distant, small and their signatures have low contrast against the background. Traditional methods based on the analysis of the difference between successive frames and a background frame will do not work. In this paper, a novel real time object detection algorithm is proposed for night-time visual surveillance. The algorithm is based on contrast analysis. In the first stage, the contrast in local change over time is used to detect potential moving objects. Then motion prediction and spatial nearest neighbor data association are used to suppress false alarms. Experiments on real scenes show that the algorithm is effective for night-time object detection and tracking.

© 2007 Published by Elsevier Ltd on behalf of Pattern Recognition Society.

**Keywords:** Visual surveillance; Night; Contrast; Detection and tracking

## 1. Introduction

Object detecting and tracking are important in any vision-based surveillance system. Various approaches to object detection have been proposed for surveillance, including feature-based object detection [1–4], template-based object detection [8,9] and background subtraction or inter-frame difference-based detection [5–7].

Background subtraction is the most popular detection method used in object trackers. In Refs. [5–7], the values taken by individual pixels over time are statistically modeled. Detection is performed by finding those pixels with values that deviate from statistical model for the background. In Ref. [6], Wren used a single Gaussian kernel to model the YUV color space of each pixel. Stauffer and Grimson [5] generalized this scheme by modeling the RGB values of pixels using a Gaussian mixture

model. There are also many feature-based motion detection methods, most of which rely on finding corresponding features in successive frames. Suitable features include Harris corners, color and contours [1,2,4].

Most algorithms for object detection and tracking are designed for daytime visual surveillance. However, night-time visual surveillance has gradually attracted more and more attention. If scene is completely dark, then it is necessary to use a thermal infrared camera [10,11]. However, the cost of a thermal camera is too high for most surveillance applications. For this reason, thermal images are not considered in this paper. It is a great challenge to detect objects at night using ordinary CCTV cameras, because the images have low brightness, low contrast, low signal to noise ratio (SNR) and nearly no color information. In this paper we focus on outdoor scenes with low light levels, just sufficient to allow a human observer to detect and track large moving objects.

Preprocessing is one way to improve image quality. Narasimhan et al. study the visual effects of different weather conditions and remove weather effects due to fog and rain [12,13]. There are also many video enhancement methods to improve image with low contrast or with low SNR images

<sup>☆</sup> Part of this paper has been announced in the ACCV06

\* Corresponding author. Tel.: +86 10 62653768.

E-mail addresses: [kqhuang@nlpr.ia.ac.cn](mailto:kqhuang@nlpr.ia.ac.cn) (K. Huang),  
[lswang@nlpr.ia.ac.cn](mailto:lswang@nlpr.ia.ac.cn) (L. Wang), [tnt@nlpr.ia.ac.cn](mailto:tnt@nlpr.ia.ac.cn) (T. Tan),  
[sjmaybank@dcs.bbk.ac.uk](mailto:sjmaybank@dcs.bbk.ac.uk) (S. Maybank).

[14,15]. Recently, Bennett has put forward an effective preprocessing method to enhance underexposed, low dynamic range video images using a dynamic function of the pixel values in spatial neighborhoods [15], but the computational cost is too high that the processing of  $640 \times 480$  video takes approximately 1 min per frame on a computer with 2.8G CPU and 512M RAM. In this paper we avoid preprocessing in order to ensure that our algorithm can run in real time.

It is clear that traditional object detection algorithm do not perform very well at night. Here we make use of some idea from human perception. In the human visual system, contrast plays an important role in detecting objects especially at night and similar to the human discrimination model-based on contrast and motion [16–19], we propose a novel object detection and tracking method for outdoor night surveillance. The method has two stages. In the first stage, objects are detected using contrast changes measured by taking sub-image inter-frame differences. In the second stage, motion prediction and spatial nearest neighbor data association are used to track objects and give feedback to the first stage. Experimental comparisons with other methods show that our method is effective for object detecting and tracking at night.

Section 2 describes the introduction of related work. Section 3 describes the new method in full. Sections 4 and 5 contain the experiments and a discussion.

## 2. Related work

There has been considerable work on feature-based, edge-based and model-based object detection and tracking, e.g. [1–3,5,8]. However, for outdoor night surveillance, the targets' low contrast and low color saturation limit the applicability of feature-based or model-based approaches. For this reason, only object detection and tracking methods based on inter-frame differences will be discussed here.

Many object detection algorithms are based on the construction of a statistical model for the values taken by background pixels. Any pixels with values incompatible with the statistical model are considered to be in the foreground. An underlying assumption of many early approaches was that a single Gaussian would be sufficient to model background pixel values. The P-finder system [6] is one example. Some simpler systems even ignore the standard deviation of the pixel values, track the mean or some other measure of central tendency and use an ad hoc threshold to assign pixels to the background or to the foreground. The simple methods for modeling the values of background pixels can fail because different objects may project to the same pixel at different times (if the objects move) and the lighting can change. Better results are obtained by using a mixture of Gaussians (MOG). Existing systems usually set the number of Gaussians within the range from 2 to 5 [5]. Furthermore, for computational reasons, the covariance matrix for each Gaussian density is assumed to be diagonal. The traditional single Gaussian model is a special case. In the MOG model, it is assumed that each pixel value satisfies a quasistationary criterion: the signal is flat fading, i.e. the change in pixel intensity value is slow compared to the update rate of

our model. MOG-based methods work well for indoor or simple urban scenes when object contrast is high, but they are not robust enough for handling outdoor objects at night.

The P-finder system [6] uses a multi-class statistical model for tracked objects, but the background model is a single Gaussian per pixel. Other papers state that the use of a single Gaussian as a background model limits robust tracking, especially with outdoor scenes containing significant clutter, e.g. [5,20]. In Ref. [5] an MOG is used for the values taken by each pixel when the pixel in question is part of the background. The parametric form of the MOG distributions can be used to classify pixels. These methods work well for indoor or simple urban scenes when object contrast is high, but they are not robust enough for handling outdoor object at night.

In the PASSWORDS project [21], the background image is continuously updated to represent the non-moving objects and scenery. An illumination change compensation algorithm ensures that the estimate of the background is stable under changes in illumination. PASSWORDS also employs a color-based shadow analysis algorithm to remove shadows. Moving objects and objects that are temporarily stationary or moving slowly are separated from the background. A similar approach is used in Ref. [22].

There are two approaches for maintaining and updating the background model: multi-sample and per-frame processing. Multi-sample approaches, e.g. [5,23], gather many samples per pixel, over many images, and then use multiple samples to compute MOG or non-parametric statistical models. The multi-sample approaches require a great deal of memory and large amounts of processing, e.g. the system in Ref. [23] required hours of computation to build the background models, and then did not update them as the scene changed.

Per-frame processing approaches update the background model for each new frame. These approaches are popular because they require much less storage and much less computation than multi-sample approaches. The main per-frame approaches are based on temporal blending or the Kalman filter, e.g. [24].

Another common method for object detection is inter-frame differencing. An object is detected if the inter-frame differences in pixel values exceed a given threshold. This method is efficient and fast, but it suffers two well-known drawbacks: foreground aperture and ghosting,<sup>1</sup> caused by frame rate and object speed [25]. To overcome these drawbacks, Cucchiara and Piccardi [26] propose a variation: the double difference which uses the thresholded differences between the frames at time  $t$  and  $t - 1$  and between the frames at time  $t - 1$  and  $t - 2$ , and combines the two differences with a logical AND. However, if an object does not have enough texture then it is not detected reliably. In the VSAM project [25], a double difference algorithm uses the difference between frames at time  $t$  and  $t - 1$  and the difference

<sup>1</sup> The output of a single difference algorithm is dependent from the speed of the moving objects. If an object moves too quickly, then the algorithm recognizes, erroneously, as foreground the pixels corresponding to the object in the old position. This problem is called ghosting. If the object moves too slowly, the foreground aperture problem occurs.

between frames at  $t$  and  $t - 2$  to erase ghosting; it also keeps in memory a background model to solve the foreground aperture problem. This system is widely used in outdoor environments with a small depth of field images, but it performs less well when the depth of field is large.

Varying lighting, small size and low contrast make night-time detection and tracking challenging [22,24,27]. We have found only a few papers within the vision and image processing community that address objects with low contrast [10,22,24]. Of these, [10,22] focus on infrared sensors and [24] is only concerned with day-time detection and tracking, which is not included in our chosen problem domain: detection and tracking at night. The primary goal of this paper is to present an algorithm for outdoor object detection and tracking at night. The algorithm is applicable to night-time images taken by a standard camera. The problems with low contrast are overcome using subimage inter-frame differences.

### 3. Object detection and tracking algorithm for night-time visual surveillance

#### 3.1. Algorithm framework

Fig. 1 shows the framework of our algorithm. The object detection algorithm includes two steps. In the first step, the object is detected using local contrast computed over the entire image. In the second step the detected objects are tracked and falsely detected objects are removed using feedback from the tracking algorithm. In Fig. 1,  $I$  is the image frame,  $R$  is the inter-frame relation which describes the similarity between two frames,  $T_1$  is the contrast threshold and  $T$  is the contrast change threshold. The details of the algorithm are given in Section 3.2.

#### 3.2. Visible image content detection based on local contrast computation

To make the detection problem more precise, let  $\{I_1 \cdot I_2 \cdots I_M\}$  be an image sequence in which each image maps a pixel coordinate  $x \in \mathcal{R}^l$  to an intensity or color  $\ell(x) \in \mathcal{R}^k$ . Typically,  $k = 1$  (e.g. gray-scale images) or  $k = 3$  (e.g. RGB color images). There are many methods to compute contrast [14,28,29]. Typically, luminance contrast is defined as the relative difference between luminance of the object,  $L_o$ , and the

surrounding background,  $L_B$ , as  $C = (L_o - L_B)/L_B$ , which is called Weber contrast [29]. Michaelson defined contrast for elementary patterns as  $C = (L_{\max} - L_{\min})/(L_{\min} + L_{\max})$  [30]. Recently more complex contrast computation methods have been proposed in the FFT and wavelet domain [14,28]. We make use of a simple measure of contrast defined as the local standard deviation  $\sigma_L$  of the image intensities divided by the local mean intensity  $\mu_L$  [31],

$$C_L = \frac{\sigma_L}{\mu_L}. \quad (1)$$

The local mean intensity  $\mu^{(p,q)}$  of a  $(2p + 1) \times (2q + 1)$  block of pixels is

$$\mu_L^{(p,q)} = \frac{1}{(2p + 1)(2q + 1)} \sum_{i=-p}^{i+p} \sum_{j=-q}^{j+q} I(i, j). \quad (2)$$

The local standard deviation  $\sigma_L^{(p,q)}$  of the block is

$$\sigma_L^{(p,q)} = \left( \frac{1}{(2p + 1)(2q + 1)} \times \sum_{i=-p}^{i+p} \sum_{j=-q}^{j+q} [I(i, j) - \mu_L^{(p,q)}(i, j)]^2 \right)^{1/2}. \quad (3)$$

The local contrast is related to entropy in a statistical sense [32], but local contrast is simpler and faster to compute.

Fig. 2 gives some examples of the statistical analysis of subimage taken from four different video sequences. The videos are “traffic sequence at night”, “two persons at night”, “sequence at day” and “infrared sequence”. (a) shows the mean of the sub-images, (b) the standard deviation and (c) the contrast. In each of (a)–(c), the horizontal axis indexes the sub-images and the vertical axis is the value of statistic. It is clear that objects can be detected when the contrast exceeds a threshold, which agrees with biological research [19]. We can see that the mean is random (from very low to very high) and is thus of little use for detection. The standard deviation can describe the local contrast in some degree [33] but it is still not appropriate for reliable object detection. The reliable detection of objects can be achieved by thresholding the contrast. The red line in (c), at 0.52, 0.31, 0.13, 0.41, respectively, is a suitable threshold in each case.

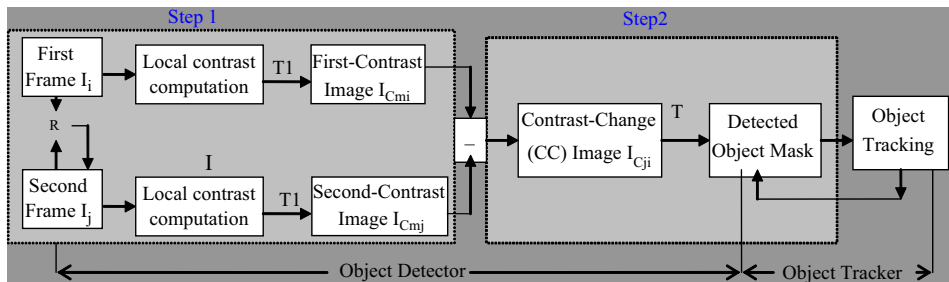


Fig. 1. Algorithm framework for object detection and tracking at night.

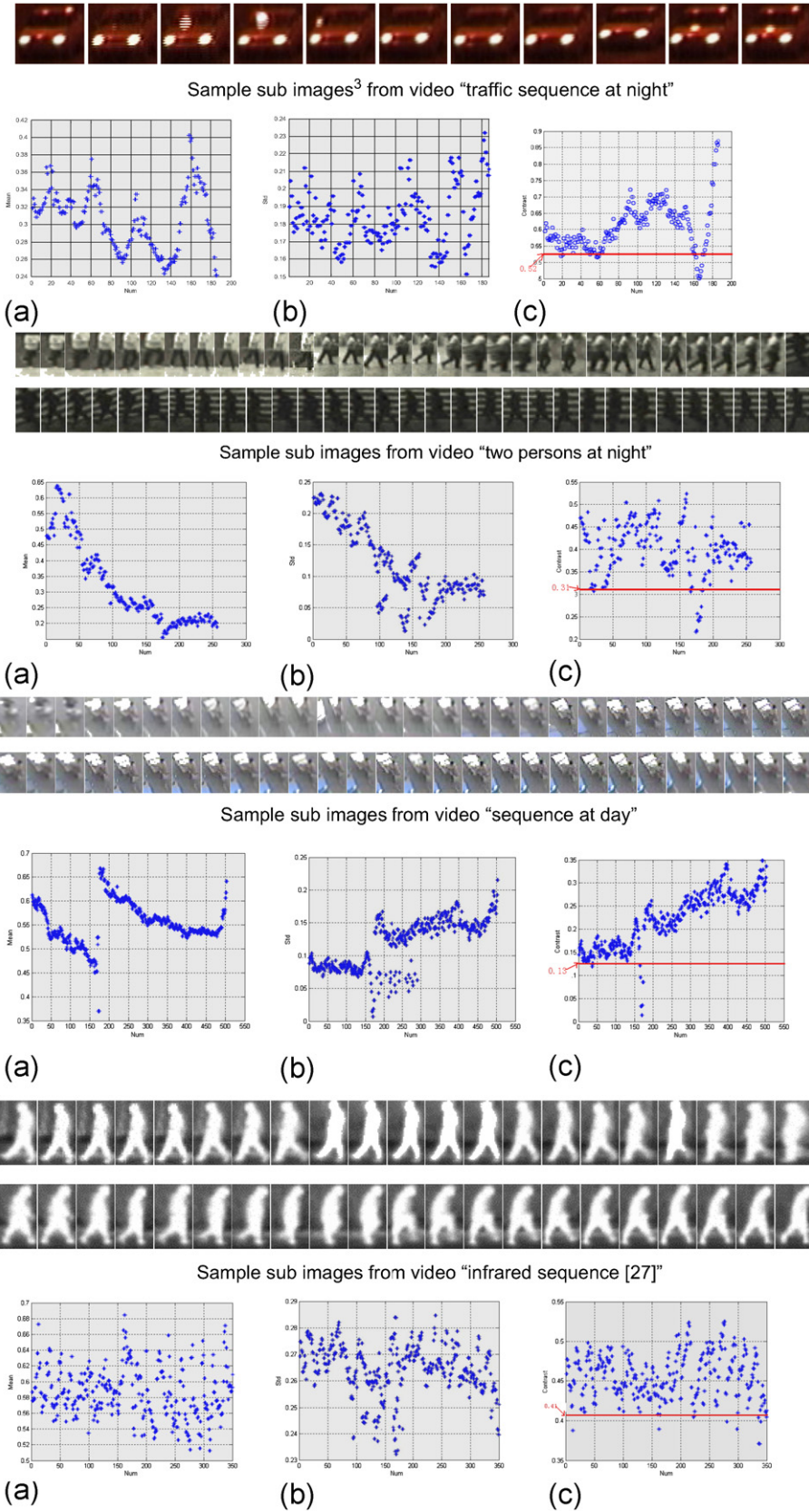


Fig. 2. Statistical analysis of four video sequences: (a) mean values for each sub-image; (b) standard deviations for each sub-image; (c) contrast distribution for each sub-image. The horizontal axes index sub-images and the vertical axes index the value of statistic in question [34].



It is apparent from the Fig. 2(c) that some sub-images containing objects have very low contrast. These sub-images cannot be found by contrast detection alone. In addition, there are some sub-images which have a high contrast but which do not contain moving objects. Examples of such sub-images are given in Table 1. For the examples in Table 1, the window size is  $44 \times 50$ . The first row contains block images chosen from frames 75–100. A human operator decides which blocks contain objects and which do not. The third, fourth and fifth rows contain the local mean, local standard deviation and local contrast values, respectively, for each block. It can be seen that the contrast of the blocks containing objects is mostly larger than some threshold (nearly 0.6), and the contrast of blocks not containing objects is usually much less, with typical values 0.4280, 0.1514. It is possible to separate the blocks containing objects from the blocks not containing objects by thresholding the contrast. It should be mentioned that the contrast of the texture part in the right-most column is also over 0.6 (equal to 0.6335), which shows that this part will also be detected. However, it is static in most of frames and can be removed by after measuring the local contrast in successive frames.

From the analysis of Fig. 2 and Table 1, we obtain the following formula to compute the mask for local contrast salience map  $M_{C_M}$ ,

$$M_{C_M}(x, y) = \begin{cases} 1 & \text{if } C^{(p,q)}(x, y) \geq T1, \\ 0 & \text{if } C^{(p,q)}(x, y) < T1, \end{cases} \quad (4)$$

$$I_{C_M} = I(x, y) \bullet M_{C_M}(x, y), \quad (5)$$

where  $C^{(p,q)}(x, y)$  is the contrast image after local contrast computation,  $I(x, y)$  is the original image and  $I_{C_M}$  shows the locations of the interested object. Threshold  $T1$  is chosen by hand. Fig. 3 shows visible object detection in a real night-time image. The window size is  $16 \times 20$  and  $T1$  is 0.45. Fig. 3(a) is original image with the size  $320 \times 240$ , Fig. 3(c) is the contrast salience map with the size of  $22 \times 12$  after local contrast computation. Fig. 3(b) is the detection result by putting local contrast salience map on Fig. 3(a). The red rectangles indicate the visible content in the image. We can see that the interested objects (people) and image structures (boundaries of illuminated areas, the waving tree) are detected by this step.

### 3.3. Moving object detection based on changing contrast saliency

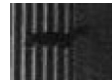
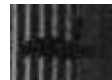


In the second step of the algorithm, changes in contrast saliency are used to filter the results of the first step and obtain the moving objects. This step is indicated by (6), (7) as follows:

$$MD_{C_{ji}}^{[p,q]} = T(|I_{C_{mi}}^{[p,q]} - I_{C_{mj}}^{[p,q]}|), \quad (6)$$

$$I_{C_{ji}}^{[p,q]} = I_i \bullet MD_{C_{ij}}^{[p,q]}, \quad (7)$$

where  $I_{C_{mi}}^{[p,q]}$  and  $I_{C_{mj}}^{[p,q]}$  are contrast images obtained from the  $i$ th image and the  $j$ th image,  $[p, q]$  is the window for contrast

Table 1  
Object detection by local contrast computation for four kinds of image (objects, black part, light part, texture part)

Images [44 × 50] (can be discriminated by eyes)													
	object	75	80	85	90	95	100	Black part 10	Light part 130	Texture part 40			
	object	0.1968	0.2064	0.2173	0.2003	0.1993	0.1731	0.1148	0.1074	0.3013			
	object	0.3251	0.3070	0.3566	0.2895	0.2778	0.2767	0.2681	0.7096	0.4755			
	object	0.6052	0.6725	0.6095	0.6919	0.7032	0.6258	0.4280	0.1514	0.6335			



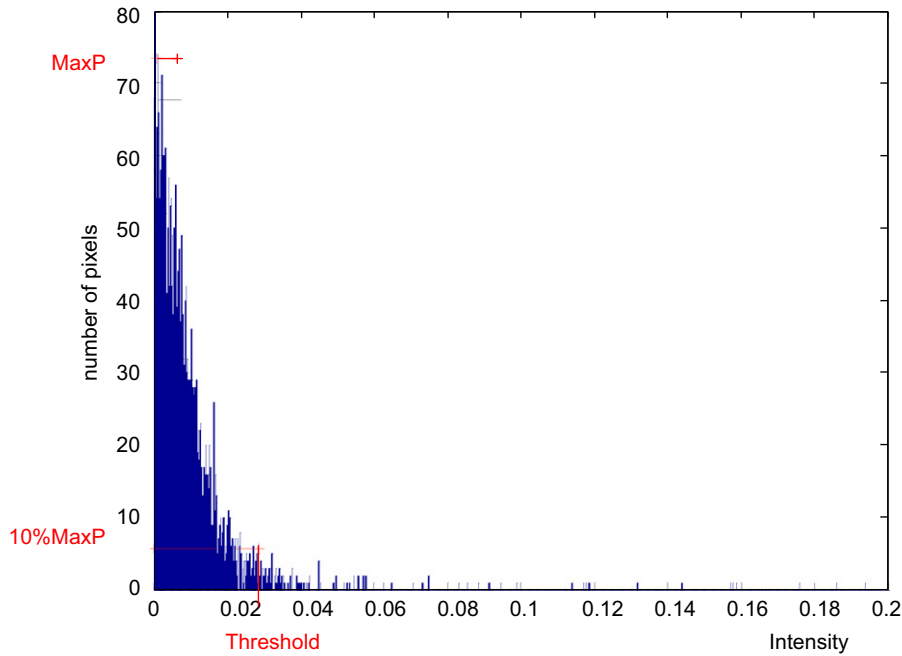


Fig. 5. Histogram of the absolute differences between the intensity values of corresponding pixels in two contrast salience images.

Experiments suggest that suitable values for  $T_R$  are in the range 0.8–0.9. The pair  $I_i$  and  $I_j$  is selected as soon as  $R(I_i, I_j)$  satisfies (9).

### 3.3.2. Adaptive threshold $T$ computation

The threshold  $T$  for contrast change is computed adaptively in each frame. It is assumed that the objects of interest occupy only a small percentage of pixels in the whole image (the assumption is often valid because we focus on outdoor surveillance with a wide field of view and in many places there are fewer moving objects at night). The histogram of the contrast difference image is used to compute  $T$ . A typical histogram is shown in Fig. 5. The algorithm for computing  $T$  is as follows.

- (1) Compute the histogram of contrast difference image  $I_{C_{ji}}$  and finding peak value of the histogram of values of pixels in the background,  $\text{Max } P = \max\{|H(k)|\}$ , where  $H(k) = \{(x, y) | I_{C_{ji}}(x, y) = k_B\}$ ,  $k_B \in [0, k_{\max}]$  is intensity value.
- (2) It is assumed that the number of pixels in an object is at least 10% of  $\text{Max } P$ . This percentage may change in other applications, depending on factors such as the number and size of observed objects, the edges in a scene and any changes in illumination.
- (3) Let  $T$  be the intensity value for which  $\text{Max } P/10 = |H(T)|$ .

### 3.4. Object tracking

An object detected by the algorithm described in Section 3.3 may overlap several local windows (blocks), for example as in Fig. 6(a). Therefore, the local windows are replaced by bounding boxes. Each bounding box is found by grouping together 8-connected clusters of blocks having similar (and non-zero)

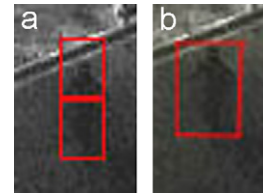


Fig. 6. (a) Local windows not grouped. (b) Local windows with 8-neighbor pairs of windows grouped together.

displacement vectors within a frame, as shown in Fig. 6(b). The grouping process is a sequential, iterative connected components labeling algorithm [29] in which the blocks having non-null displacement play the role of “foreground points” and two blocks are considered neighbors if they are 8-connected.

Once the object areas are determined in each frame, the tracking algorithm is needed to trace the objects from frame to frame.

Object tracking methods can be classified into model-based, appearance-based, contour and mesh-based, feature-based, and hybrid methods [35]. None of these methods work well at night. The tracking algorithm used in this work is based on an object’s position and velocity.

$$\tilde{P}_i(x, y) = \tilde{P}_{i-1}(x, y) + \tilde{V}_i(x, y) \cdot \left(\frac{N}{2} + 1\right), \quad (10)$$

$$\tilde{V}_i(x, y) = \frac{\sum_{k=1}^N \tilde{P}_{i-k}(x, y) - \sum_{k=M+1}^{M+N} \tilde{P}_{i-k}(x, y)}{M \times N}, \quad (11)$$

where  $\tilde{P}$  and  $\tilde{V}$  are the estimated object position and velocity, respectively. The values of  $M$  and  $N$  are chosen by hand. Suitable values are  $N = 5$ ,  $M = 11$ . The data association

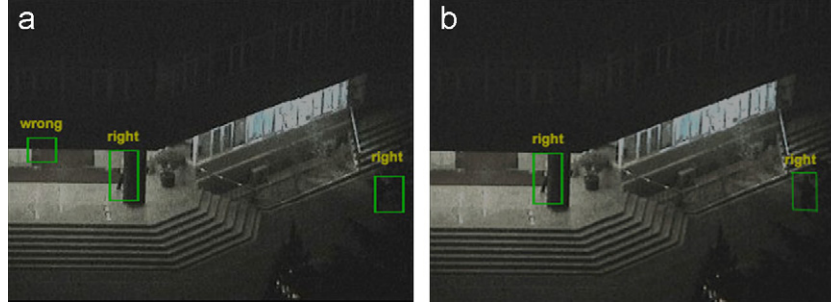


Fig. 7. Example in which a falsely detected object is filtered out: (a) false object detection caused by a large gray level gradient and (b) the falsely detected object is filtered out by multi-frame matching.



Fig. 8. NLPR surveillance system.

can be realized by Mahalanobis distance

$$D_{i,i-1} = (\tilde{P}_i(x, y) - \tilde{P}_{i-1}(x, y))^T \times \Sigma^{-1}(\tilde{P}_i(x, y) - \tilde{P}_{i-1}(x, y)), \quad (12)$$

where the  $D$  is 50 pixels, which is also given manually.

Multi-frame matching (here two frames) is used to filter out the falsely detected objects as in Fig. 7(a) caused by large gray level gradients. (The problem with “light variations” is that it might mean a variation in lighting from one image to the next.) This leads to more accurate detection results, for example, as shown in Fig. 7(b).

#### 4. Experimental results and discussion

In this section, the results of proposed algorithm for night-time object detection and tracking are assessed. All the night scene videos are captured by standard CCD cameras (Panasonic WV-CW860A), with a frame size of  $320 \times 240$  pixels. The cameras are part of the NLPR (National Laboratory

of Pattern Recognition) surveillance system. The surveillance system consists of 19 cameras including several PTZ cameras. In our experiments, only the intensity value is used as the input. The effectiveness of the proposed method is verified by the detection results and tracking results. The experiments also verify the low computational cost of the algorithms (Fig. 8).

##### 4.1. Algorithm testing

###### 4.1.1. Object detection

The object detection algorithm is tested on the sequence “two person at night” from frame 1 to 120. Fig. 9 gives the detection results for frames 1–120. The first column is the local contrast computation result for frames 1, 20, 40, 60, 80, 100, 120. The moving objects and other visible content are detected in this step. The second column is the contrast change result between the successive frames 1, 20, 40, 60, 80, 100, 120. The changes are detected. All the detection results from frame 1 to 120 are plotted on one frame as the third column in Fig. 9. It is clear that the moving objects are accurately located in all the frames.



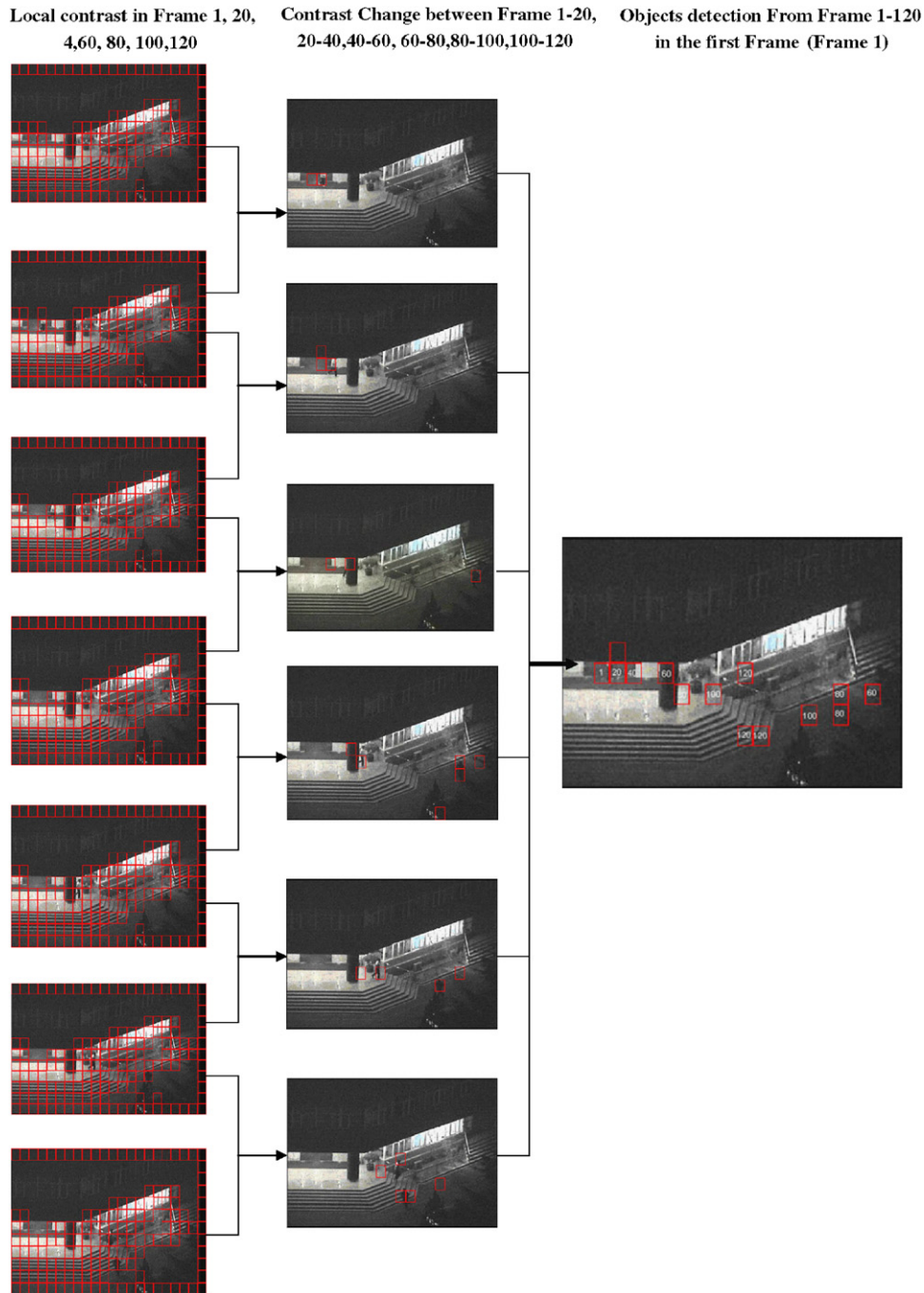


Fig. 9. Detection results for frames 1–120. First column: visible objects found by local contrast computation. Second column: interesting objects detected by changing contrast saliency. Third column: consecutive detection results from frame 1 to 120 on the first frame (frame 1).

#### 4.1.2. Comparison with other methods for object detection

The object detection algorithm is compared with two background-based object detection algorithms in Fig. 10. One of the algorithms uses an adaptive MOG model [5], the other uses nonparametric background subtraction (NPB) [24]. Fig. 10(a) is the original image. This image is low contrast and there are some distractors such as the light from the windows and the tree. Fig. 10(b) shows the detection results obtained from the MOG algorithm. The two objects are detected but not very well even though the parameters in the MOG algorithm

are carefully chosen, especially the threshold  $T$  and standard deviation  $\sigma$ . Fig. 10(c) shows the detection result using NPB. The results are better than those obtained from the MOG algorithm. Most of the pixels associated with the two objects are found. This algorithm is affected by high gray level gradients. For example, there are some false detections near the windows. Fig. 10(d) shows the detection results obtained using the new algorithm. The block size is  $2 \times 2$ . It can be seen that the new algorithm performs better than MOG and NPB with more accurate detections and fewer false alarms.

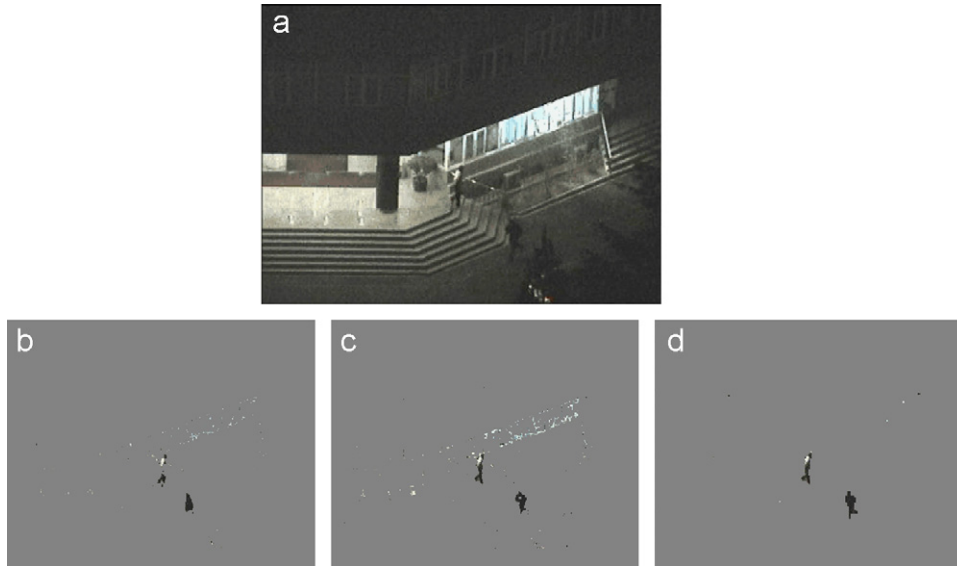


Fig. 10. Comparisons of three object detection algorithms: (a) original image 110# in sequence “two persons at night”, (b) result of MOG, (c) result of NPB, and (d) result of our algorithm.

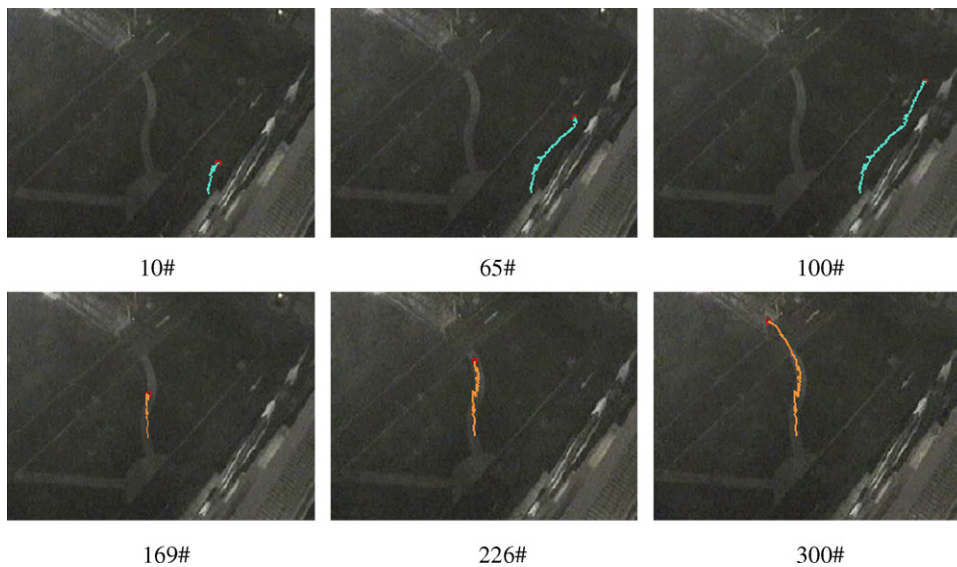


Fig. 11. Object detection and tracking results at night.

The algorithm was tested in more cluttered night-time scenes. The camera is wide view, the objects have low contrast and a very small apparent size ( $2 \times 3$  pixels). Most of the current algorithms, including NPB, fail to detect the objects. Fig. 11 shows the detection and tracking results. There are two objects in the video. The first object is tracked from frame 10 to 100 as shown by the green trajectory and the second object is tracked from frame 169 to 300 as shown by the orange trajectory. The second object is successfully tracked even when partially occluded by the shed about frame 300.

#### 4.1.3. Reflection and shadow processing

Light reflections are very prominent in many night-time scenes, for example traffic scenes. Our algorithm can detect

objects while ignoring the reflections. It is tested on a traffic sequence and the results compared with those obtained from an MOG algorithm.<sup>2</sup> For sequences with moderate contrast, MOG and NPB can detect objects, and the nonparametric algorithm can remove shadows by considering color information. However, color information cannot be reliably obtained at night. We just compared our algorithm with the MOG algorithm. Fig. 12(a) shows the results using MOG. The object detection is adversely influenced by reflections of the car headlights. While the detection and tracking results by our

<sup>2</sup> The object detection is based on MOG background subtraction. The tracking algorithm described in Section 3.4 is used. The parameters in MOG algorithm have been adjusted to get the best result.

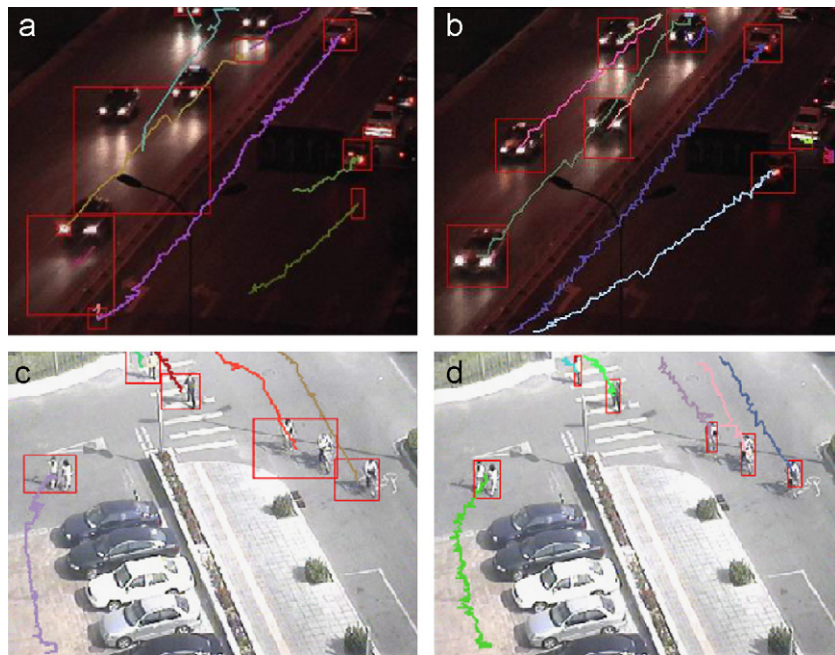


Fig. 12. Comparisons of experiment results from the MOG-based algorithm and from the new algorithm: (a) result obtained from MOG, (b) result obtained from our algorithm, (c) result obtained from MOG [5], and (d) result obtained from our algorithm.

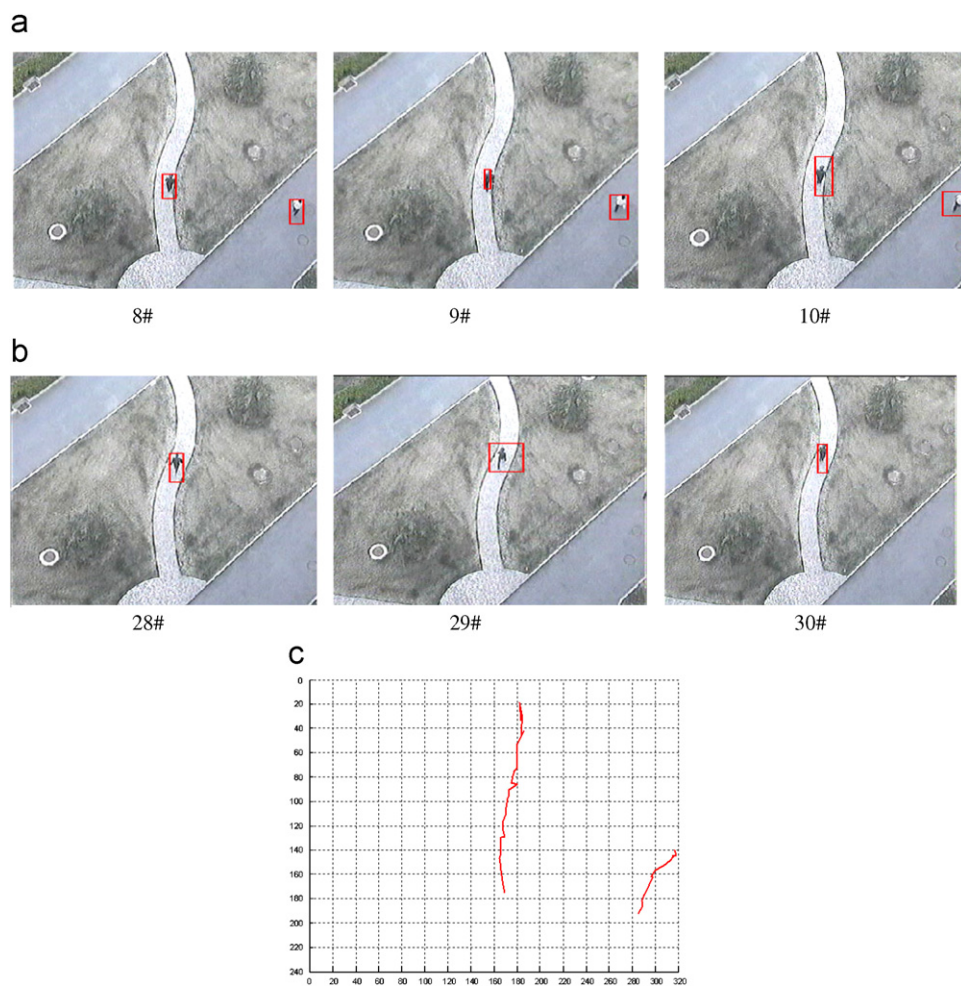


Fig. 13. Object detection and tracking during camera zooming: (a) three frames taken during zooming in; (b) three frames taken during zooming out; and (c) trajectories of objects in this sequence as the camera first zooms in and then zooms out.



Table 2

Comparisons between the new algorithm and the MOG-, NPB-based algorithms on the different sequences of frames

Normalized Jaccard coefficient $NJ = \sum_n J/n$ (%)	Our method (%)	MOG	NPB
“Two persons at night”	96.68	78.65%	85%
“Traffic sequence at night”	96.42	55.71%	63.98%
“Persons at a distance”	90.12	Low than 10% (ignored)	Low than 10% (ignored)
“Zooming camera sequence”	94.44	Low than 10% (ignored)	Low than 10% (ignored)

algorithm is good, it is that the local contrast is low for the light reflected by the road. For the same reason, our algorithm can also work well for shadow removing in daytime as (d) shows.

#### 4.1.4. Object detection with zooming camera

Our algorithm is also robust to camera zooming. Fig. 13(a) shows the detection results obtained using three consecutive frames taken by a camera zooming in and (b) shows the detection results for the camera zooming out over three consecutive frames. The objects are detected accurately. Fig. 13(c) shows the trajectories of the objects when the camera first zooms in and then zooms out. It is clear that objects can be tracked robustly during camera zooming. The attachments contain video sequences showing the results.

#### 4.2. Evaluation of the experiments

Comparisons are made between the new algorithm and the MOG- and NBP-based algorithms. The performances of the three algorithms are measured using the Jaccard coefficient [36],  $J$  defined by

$$J = TP / (TP + FP + FN) \%, \quad (13)$$

where TP (true positives) is the number of moving objects correctly detected, FP (false positives) is the number of false detections of moving objects and FN (false negatives) is the number of missed detections of moving objects. The ground truth is obtained manually in each frame of the sequence.

Table 2 gives the values of normalized Jaccard (NJ) coefficient in Table 2,  $J$  is used to describe the detection accuracy of single frame and NJ gives the detection accuracy for the whole sequence. For the sequence “two persons at night”, the MOG and NBP algorithms both succeed to some extent. The NBP algorithm detects low contrast objects better than the MOG algorithm but it has too many false alarms. The new algorithm achieves the best (96.68%). For the sequence “traffic sequence at night”, the MOG and NBP algorithms do not work well because of the light reflections. The new algorithm gives satisfactory results (96.42%). For the two sequences “persons at a distance” and “zooming camera sequence”, it is very difficult to detect objects using the MOG and NBP algorithms because of the low contrast and the zooming camera. The NJ score is so low that it is ignored in Table 2. The new algorithm works well on the two sequences (90.12% and 94.44%).

The new algorithm can run in real time. After the local contrast computation step, the image sizes are reduced to  $[320/M, 240/N]$ , where  $[M \times N]$  is the local window size. In

our experiments, the local window size is  $4 \times 4$ , which ensures that the algorithm can run real time (more than 25 FPS at  $320 \times 240$  resolution with a 1.8G CPU).

#### 5. Conclusion and future work

Object detection and tracking at night is very important for night surveillance, which is key part of 24 h visual surveillance. In this paper, we proposed an object detection and tracking algorithm for night surveillance based on inter-frame differences. Object detection is based on local contrast changes and detection results are improved by tracking the detected objects from one frame to the next. Experiments demonstrate that our algorithm has the ability to detect and track objects robustly at night under conditions in which more conventional algorithms fail. There are several parameters and thresholds in the new algorithm. Some parameters are adjusted adaptively, for example, the threshold to determine significant inter-frame differences and the threshold on the differences between contrast scores. Other parameters such as the size of rectangular region for contrast measure, the threshold on contrast measure and the threshold on the distance measure between two rectangles are chosen by hand. In the future, we will use a multi-scale algorithm, similar to those used in face detection, to decide the size of the rectangular region for contrast measure. The threshold on contrast measure will be decided by a learning algorithm. These methods for computing thresholds automatically will improve our system greatly.

#### Acknowledgments

This work is supported by National Key Basic Research Projects of China 973 Grant (2004 CB312102), National Science Founding (60605014, 60332010 and 60335010) and CA-SIA Innovation Fund for Young Scientists. The authors also thank the anonymous reviewers and the editor for their valuable comments.

#### References

- [1] D. Crandall, J. Luo, Robust color object detection using spatial-color joint probability functions, in: Proceedings of CVPR, 2004, pp. 379–385.
- [2] J. Weng, J. Ahuja, N. Huang, Matching two perspective views, PAMI 14 (8) (1992) 806–825.
- [3] W. Krattenthaler, K.J. Mayer, M. Zeiler, Point correlation: a reduced const template matching technique. ICIP, 1994, pp. 208–212.
- [4] C.G. Harris, M. Stephens, A combined corner and edge detector, Fourth Alvey Vision Conference, 1988, pp. 147–151.



- [5] C. Stauffer, W.E.L. Grimson, Adaptive background mixture models for real-time tracking, *CVPR*, 1999, pp. 252–260.
- [6] C.R. Wren, A. Azarbayejani, A. Pentland, Pfunder: realtime tracking of the human body, *PAMI* 19 (7) (1997) 780–785.
- [7] L. Liyuan, W. Huang, et al., Statistical modeling of complex backgrounds for foreground object detection, *IEEE Trans. Image Process.* 13 (11) (2004) 1459–1472.
- [8] A.K. Jain, Y. Zhong, S. Lakshmanan, Object matching using deformable templates, *PAMI* 18 (3) (1996) 267–278.
- [9] S. Sclaroff, L. Liu, Deformable shape detection and description via model based region grouping, *PAMI* 23 (5) (2001) 474–489.
- [10] D. Davies, P.L. Palmer, M. Mirmehdi, Detection and tracking of very small low contrast objects, *BMVC*, 1998, pp. 599–608.
- [11] Second Joint IEEE International Workshop on Object Tracking and Classification in and Beyond the Visible Spectrum (OTCBVS'05) (<http://www.cse.ohio-state.edu/otcbvs/>).
- [12] S.G. Narasimhan, S.K. Nayar, Contrast restoration of weather degraded images, *PAMI* 25 (6) (2003) 713–724.
- [13] K. Garg, S.K. Nayar, Detection and removal of rain from videos, in: *Proceedings of IEEE Computer Vision and Pattern Recognition (CVPR)*, Washington, 2004, pp. 528–535.
- [14] J. Tang, J.H. Kim, E. Peli, Image enhancement in the JPEG domain for people with vision impairment, *IEEE Trans. Biomed. Eng.* 51 (11) (2004) 2013–2023.
- [15] E.P. Bennett, L. McMillan, Video enhancement using per-pixel virtual exposures, *SIGGRAPH2005*, 2005.
- [16] L. Itti, C. Koch, Computational modeling of visual attention, *Nature Neurosci. Rev.* 2 (3) (2001) 194–203.
- [17] C. Koch, S. Ullman, Shifts in selective visual attention: towards the underlying neural circuitry, *Hum. Neurobiol.* 22 (1985) 219–227.
- [18] V.S. Vaingankar, V. Chaoji, R.S. Gaborski, A.M. Teredesai, Cognitively motivated habituation for novelty detection in video, *NIPS Workshop on 'Open challenges in cognitive Vision'*, Whistler, BC, Canada, December, 2003.
- [19] J. Kaas, C.E. Collines, *The Primate Visual System*, Social Science, 2003.
- [20] I. Haritaoglu, D. Harwood, L. Davis, W<sup>4</sup>s: a real-time system for detecting and tracking people in 2.5D, *ECCV*, 1998.
- [21] M. Bogaert, N. Chleq, P. Cornez, C. Regazzoni, A. Teschioni, M. Thonnat, The passwords project, *ICIP*, IEEE, 1996, pp. 1112–1115.
- [22] G. Foresti, Object detection and tracking in time-varying and badly illuminated outdoor environments, *SPIE J. Opt. Eng.* 37 (9) (1998) 2550–2564.
- [23] S. Rowe, A. Blake, Statistical background modeling for tracking with a virtual camera, *Proceedings of British Machine Vision Conference*, 1995.
- [24] A. Elgammal, R. Duraiswami, D. Harwood, L.S. Davis, Background and foreground modeling using non-parametric kernel density estimation for visual surveillance, *Proceedings of the IEEE*, July 2002.
- [25] R.T. Collins, A.J. Lipton, T. Kanade, et al., A system for video surveillance and monitoring, Technical Report, Carnegie Mellon University, 2000.
- [26] R. Cucchiara, M. Piccardi, Vehicle detection under day and night illumination, *Proceedings of the Third International Symposium on Intelligent Industrial Automation and Soft Computing*, 1999.
- [27] T.E. Boult, R.J. Micheals, X. Gao, M. Eckmann, Into the woods: visual surveillance of non-cooperative camouflaged targets in complex outdoor settings, *Proceedings of the IEEE*, October 2001, pp. 1382–1402.
- [28] E. Peli, Contrast sensitivity function and image discrimination, *J. Opt. Soc. Am. A* 18 (2) (2001) 283–293.
- [29] R.M. Haralick, L.G. Shapiro, *Computer and Robot Vision*, vol. 1, Addison-Wesley, Reading, MA, 1992 pp. 28–33.
- [30] P.G.J. Barten, Contrast sensitivity of the Human Eye and its Effects on Image Quality, *SPIE*, Bellingham, Washington, 1999.
- [31] E. Reinhard, P. Shirley, M. Ashikhmin, T. Troscianko, Second order image statistics for computer graphics, *ACM Symposium on Applied Perception in Computer Graphics and Visualization*, August 2004.
- [32] Y.-F. Ma, H.-J. Zhang, Detection motion object by spatial-temporal entropy, *ICME*, 2001, pp. 381–384.
- [33] K. Huang, Z.-Y. Wu, Q. Wang, Image enhancement based on the statistics of visual representation, *Image Vision Comput.* 23 (2005) 51–57.
- [34] (<http://www.cse.ohio-state.edu/OTCBVS-BENCH/bench.html>).
- [35] A. Cavallaro, O. Steiger, T. Ebrahimi, Multiple video object tracking in complex scenes, *Proceedings of ACM Multimedia*, Juan les Pins, France, 2002, pp. 1–6.
- [36] P. Sneath, R. Sokal, *Numerical Taxonomy. The Principle and Practice of Numerical Classification*, W.H. Freeman, New York, 1973.

**About the Author**—KAIQI HUANG received his M.Sc. from Nanjing University of Science and Technology in 2001 and obtained his Ph.D. degree from Southeast University in 2004. During May 2004–December 2005 he was a postdoctoral fellow in NLPR, institute of Automation, Chinese Academy of Science, China. Since December 2005, he has been an associate professor in NLPR. His current research interests include visual surveillance? digital image processing, pattern recognition. He is member of IEEE and Deputy secretary of IEEE Beijing Section, local chair of the fifth international visual surveillance workshop in conjunction with ICCV05 and committee member of the sixth international visual surveillance workshop in conjunction with ECCV06. He has published nearly 20 papers on major international journals and conferences, such as computer vision and image understanding, image and vision computing and ICPR.

**About the Author**—TIENIU TAN received the B.Sc. degree in electronic engineering from Xi'an Jiaotong University, China, in 1984 and the M.Sc., DIC, and Ph.D. degrees in electronic engineering from Imperial College of Science, Technology and Medicine, London, UK, in 1986, 1986, and 1989, respectively. He joined the Computational Vision Group, Department of Computer Science, The University of Reading, England, in October 1989, where he worked as Research Fellow, Senior Research Fellow, and Lecturer. In January 1998, he returned to China to join the National Laboratory of Pattern Recognition, the Institute of Automation of the Chinese Academy of Sciences, Beijing. He is currently Professor and Director of the National Laboratory of Pattern Recognition as well as President of the Institute of Automation. He has published widely on image processing, computer vision and pattern recognition. His current research interests include speech and image processing, machine and computer vision, pattern recognition, multimedia, and robotics. Dr. Tan serves as referee for many major national and international journals and conferences. He is an Associate Editor of Pattern Recognition and of the IEEE Transactions on Pattern Analysis and Machine Intelligence, as the Asia Editor of Image and Vision Computing. He was an elected member of the Executive Committee of the British Machine Vision Association and Society for Pattern Recognition (1996–1997) and is a founding co-chair of the IEEE International Workshop on Visual Surveillance.

**About the Author**—LIANGSHENG WANG received his B.Sc. in electrical engineering and M.Sc. in video processing & multimedia communication from Beijing Traffic University, China, in 2001 and 2004, respectively. He is currently a Ph.D. candidate in pattern recognition & intelligent system in the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His current research interests include computer vision, pattern recognition, digital image processing and analysis, multimedia, visual surveillance, etc.

**About the Author**—STEVE MAYBANK received a B.A. in Mathematics from King's College Cambridge in 1976 and a Ph.D. in Computer Science from Birkbeck College, University of London in 1988. He joined the Pattern Recognition Group at Marconi Command and Control Systems, Frimley in 1980 and moved to the GEC Hirst Research Centre, Wembley in 1989. In 1993–1995 he was a Royal Society/EPSC Industrial Fellow in the Department of Engineering Science at the University of Oxford. In 1995 he joined the University of Reading as a Lecturer in the Department of Computer Science. In 2004 he became a Professor in the School of Computer Science and Information Systems, Birkbeck College. His research interests include the geometry of multiple images, camera calibration, visual surveillance, information geometry and the applications of statistics to computer vision. He is a member of the IEEE, a Fellow of the Royal Statistical Society, a Fellow of the Institute of Mathematics and its Applications, a member of the British Machine Vision Association and a member of the Societe Mathematique de France.