

Unobtrusive silhouette extraction using multivariate analysis and shadow removal in RGB color model

Mohit Kumar Singh¹ · Neeta Nain¹

Received: 31 May 2016 / Accepted: 3 October 2016 / Published online: 21 October 2016
© Springer International Publishing Switzerland 2016

Abstract Object extraction is the initial step for any image processing task, such as visual surveillance, object tracking and intelligent transport system. Apart from other techniques, such as biometric system, these methods do not require any physical contact with an individual and can be performed from a distance and, therefore, used severely. Although several object extraction methods are available, this paper proposes a simple method to extract an object from video sequence based on the correlation between color model components. As the use of a single threshold is not suitable to correctly segment the object from the background, we are using two thresholds. Experimental results prove that this method can work well in static as well as dynamic background having different illumination conditions. Moreover, segmentation has side effects like moving cast shadows and self shadow, both of which reduce accuracy. In this paper, we present an automatic shadow removal method to extract objects. The analytical comparison with other foreground extraction, shadow detection and removal algorithms and their results are also presented for better understanding. The proposed method extract silhouette without any pre or post processing step of shadow removal.

Keywords MANOVA · Chi-square distribution · Threshold · Multiband · Trivariate · Autocorrelation · Radiometric similarity

1 Introduction

Video object segmentation that is used primarily in various application areas of computer vision which include video surveillance system, artificial intelligence enabled traffic monitoring system, path detection, robotics, autonomous navigation, activity based human recognition and many other. Surveillance is one of the most critical areas which include detection, tracking, classification of moving object or group of objects and recognition of various motions or pose. The effectiveness of these systems is characterized by first how accurately in shape and size the system can detect the object or any suspicious behavior of an object (human), and second by how reliable the system is in different environmental conditions, such as lighting and background conditions.

In computer vision, sensors are used to capture the real-time picture or scenario. The images generated, can contain several moving or stationary objects in front of a static or dynamic background. A common assumption is that the background is of static nature since in many surveillance systems the camera remains fixed. Some researchers [1–3] have shown that a background only changes due to the motion of the camera and can be compensated overall. But there are situations where we have continuous motion in the background such as tree leaves, water bodies or some moving object which becomes stationary later on.

Therefore, instead of considering our background static, the foreground object is the set of pixels that is not stationary and changes its position and direction between frames. Moving foreground object detection can be achieved by two different ways: (1) by motion detection and (2) by motion estimation. In detection, we identify changed regions from the frames when the camera is fixed. Motion estimation estimates the motion vector or the expected position of the moving object in next frame. Sometimes in surveillance, it

✉ Mohit Kumar Singh
2014pcp5352@mnit.ac.in

Neeta Nain
nnain.cse@mnit.ac.in

¹ Malaviya National Institute of Technology, Jaipur, India

is required to find out the speed, acceleration of the moving object.

In the practical situation, object extraction becomes tough. The presence of noise generated from the capturing device, textural similarity between foreground and background, change in lighting conditions. Dynamic background (water bodies, tree leaves, rain, wind), some moving object which become static after moving or starts moving suddenly, occlusion between objects and last but not least presence of shadow.

There are several approaches for foreground object extraction such as temporal differencing, spatial homogeneity, optical flow, and change detection [1, 4, 5] which can be pixel based or region based. Pixel based detection algorithms are sensitive to pixel variations (noise, illumination changes). Methods [6–8] handles noise and light change using adaptive background model. Tsai and Lai [9] do not use background model and instead analyze independent components. On the other hand, region based change detection method measures the characteristic of a region at some pixel location. Likelihood ratio test [10] uses the hypothesis test to check the intensity distribution of a region. Shading model [11] considers the ratio of intensity in a region. Liu et al. [12] considers the reflectance component of image intensity. Li and Leung [13] combines texture and intensity difference.

The requirement of a reference frame (frame with no object) is the one that restricts the use of change detection algorithm. Cases, where there is a difference in the speed of the moving object and the cases where the object in the frame moves and stop for some time, makes the identification difficult with change detection algorithm. Also, the availability of cast shadow in the background region can cause problems in detection.

Among all the above approaches, the widely used approach in the absence of any available knowledge of foreground object or the background for object detection with fixed camera is background subtraction [6, 7, 14–20]. A background reference frame is computed just by averaging the background frames or by using an initial estimation of background frame and the iteratively updating it to obtain the final estimate. Pfister [6] uses Gaussian distribution at each pixel as the background model. Haritaoglu et al. [14], models the background by representing pixels by minimum, the maximum of intensity and maximum intensity difference between frames. Marko and Matti [21] present a texture based method that each pixel is modeled as a group of adaptive local binary histogram. The background model should be able to reflect the real background as accurately as possible and should reflect with sudden scene change such as start or stop of a moving object. Also detection of ghost and shadow effects the detection of an object.

Whether the object detection method used is pixel based or region based, thresholding the difference image is the

most challenging task. In many cases, a single threshold is used, but the problem with this approach is that single threshold is enough to separate two classes. From classification point of view, applying P thresholds result in $P + 1$ classes. For $P = 1$, we have two categories as background and foreground. Consider a histogram of pixel intensities of a given frame (background plus foreground). In the ideal case (bimodal distribution), the histogram has a deep and sharp valley between two peaks (representing object and background). In this case, a single threshold T_1 is enough to separate two classes. However, in real cases, the valleys are flat, broad and noisy, and it is tough to obtain sharp valleys making it difficult to find the threshold value to segment. Instead of selecting a threshold by trial and error several adaptive algorithms [22–29] are proposed. To overcome the problem with single threshold it is better to consider multiple thresholds.

In this paper, our aim is to detect a moving object with high accuracy by reducing the False Negative and False positive as much as possible. The organization of the paper is as follow. Section 2 explains the algorithm for object extraction. Section 3 explains Multivariate analysis of variance using Chi-square distribution. Section 4 explains hypothesis generation for object detection. Section 5 provides experimental results and analysis. Section 6 describes shadow detection and removal approaches. And the conclusion is given in Sect. 7.

2 Proposed algorithm for object detection

The basic idea of our algorithm is change detection. However, the moving object region is not obtained directly by background subtraction. In other words, our estimation of background is based on multivariate analysis of data. The multivariate analysis consists of methods that we can use when several measurements are performed on each object in one or more samples. The measurements are known as variables, and the individuals or objects are the units (research units, sampling units, or experimental units) or observations. Some real world examples of multivariate data units and their variables are given in Table 1. Similarly, we can deduce that the RGB image which we are using for object extraction is a multivariate data unit, and the variables are R, G and B color components respectively.

Table 1 Examples for multivariate units and variables

Units	Variables
Students	Marks of various subjects
People	Height, weight, heart rate
Bank loans applicants	Income, education, savings
RGB image	R, G and B components

Sometimes, it is wise to extract each variable available and study them separately. But they may be correlated in nature with other variables. Thus, in many cases, the variables are tangled in such a way that study of individual may not provide enough information. Multivariate analysis, provide us the methods to examine the behavior of correlated variables simultaneously so that we can access the key features of the process that produced them.

The multivariate methods help us to (1) find out the joint performance of the M.V variables and (2) to identify the effect of one on the other. The multivariate analysis provides both descriptive and inferential procedures in which we can search for patterns in the data or test hypotheses about patterns. Several methods are available that focuses primarily on variance, covariance, ratio of variance. The most commonly used methods MANOVA and ANOVA deals with the variables variance. Variance is a numerical representation of the distribution of variables in the population. If two variables are associated or correlated with one another, then they share some common property that makes them vary together.

This concept of multivariate analysis can easily be extended to extract the foreground object. The input images (containing the moving object) are correlated M.V units and the color components, which in our case are R, G and B components acts as M.V variables. A block diagrammatic representation of the proposed scheme is shown in Fig. 1. The complete process is divided into five major steps. The first step is to generate a background model. In our algorithm, we are using very simple method for background estimation which is averaging the background. The second phase is to generalize the M.V variables (R, G, B color components) as multivariate Gaussian distribution. Step third uses MANOVA and Chi-squared distribution to identify the correlation between the variables which is further used in Step fourth to generate the hypothesis. And finally in step five, the input image pixels are verified if they satisfy the generated hypothesis or not. Each step is further explained in the following subsections.

2.1 Multivariate Gaussian generalization for RGB Color components

Multivariate procedures will be based on multivariate normal distribution and have some basic properties as below:

- The distribution can be completely described using only means, variances and covariance.
- If the variables are uncorrelated, they are independent.
- The dependent variable should be normally distributed within groups.

Salvador et al. [30] states that if we can find a unit vector (r^*, g^*, b^*) in the *RGB* space and project each pixel color

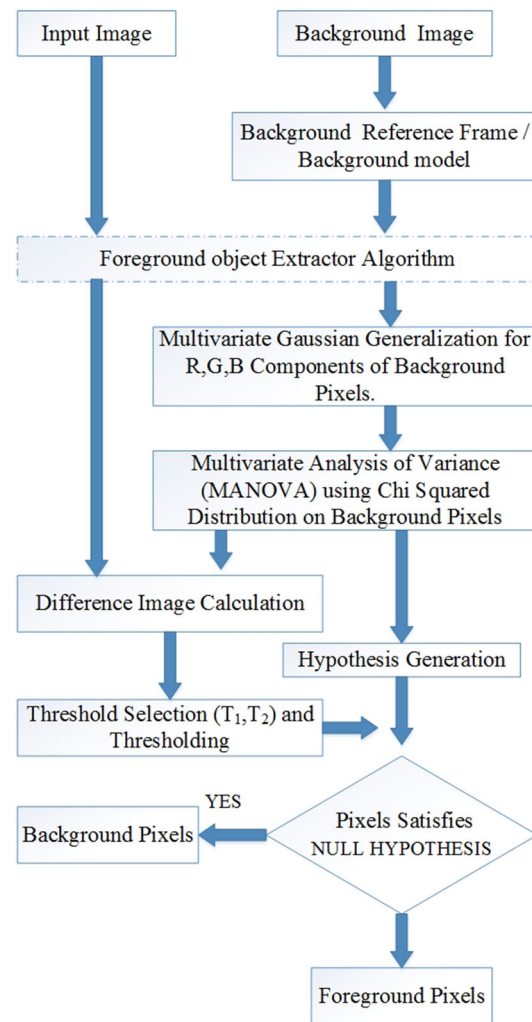


Fig. 1 Block diagram of proposed object extraction method

vector $(R, G, B)_{(x,y)}$ onto this vector. The projected length is the intensity $I_{(x,y)}$ and the residual vector is perpendicular to the color vector and lies on a 2D plane β . By analyzing the distribution of residuals in plane β it is found that the residuals can be modeled by a 2D normal distribution and its iso-value curve can be represented by an ellipse in the plane.

Based on these observations, considering R, G and B components as the random variables with their respective mean and variance. The Multivariate generalization in d -dimensional space ($d = 3$) is given by:

$$p(x) = \frac{1}{(2.\pi)^{\frac{d}{2}} |\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right) \tag{1}$$

where, $\mu = E[x]$ is the mean value and σ is the $(d \times d)$ covariance matrix given as,

$$\text{Cov}(x_1, x_2, x_3) \text{ or } \Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_2^2 & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_3^2 \end{bmatrix} \tag{2}$$

Considering the case of diagonal covariance matrix, the iso-value curves is equivalent to,

$$x^T \Sigma^{-1} x = [x_1, x_2, x_3] \begin{bmatrix} \frac{1}{\sigma_1^2} & 0 & 0 \\ 0 & \frac{1}{\sigma_2^2} & 0 \\ 0 & 0 & \frac{1}{\sigma_3^2} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = C \quad (3)$$

$$\frac{x_1^2}{\sigma_1^2} + \frac{x_2^2}{\sigma_2^2} + \frac{x_3^2}{\sigma_3^2} = C \quad (4)$$

This is the equation of an ellipse whose axes are determined by the the variances of the involved features. In our case, with three features R , G and B above equation changed as

$$\left(\frac{x_R}{\sigma_R}\right)^2 + \left(\frac{x_G}{\sigma_G}\right)^2 + \left(\frac{x_B}{\sigma_B}\right)^2 = C \quad (5)$$

The distribution of RGB components (blue colored samples) in the background model is shown in Fig. 2.

3 Multivariate analysis of variance using chi squared distribution

In the above Eq. (5), ‘ C ’ defines the scale of the ellipse and could be any arbitrary number. The question is now how to choose C , such that we can represent an ellipse with a given confidence level (e.g. 95 or 99 %). The left-hand side of Eq. (5), is the sum of the square of normally distributed random variables. Chi-square distribution is known to be suitable here, which states that let $x_i, i = 1, 2, \dots, N$, be samples of a gaussian distribution then, y is a chi-square distributed variable with N degree of freedom.

$$y = x_1^2 + x_2^2 + \dots + x_N^2 \quad (6)$$

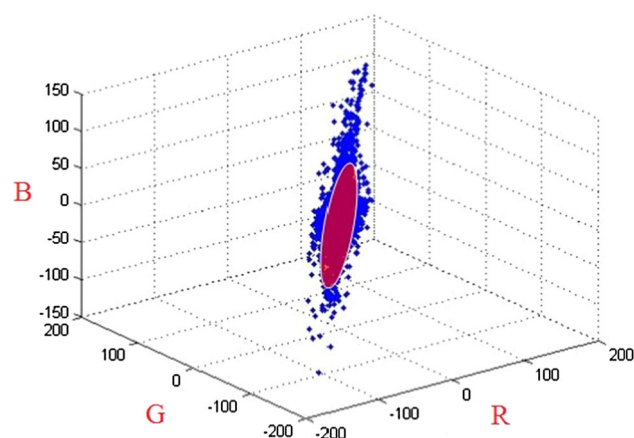


Fig. 2 Representation of RGB distribution and 99 % confidence ellipse obtained using Chi-squared distribution

Table 2 Table for Chi-square probability

df	Probabilities						
	0.99	0.975	0.95	...	0.05	0.025	0.01
1	–	0.001	0.004	...	3.841	5.024	6.635
2	0.020	0.051	0.103	...	5.991	7.378	9.210
3	0.115	0.216	0.352	...	7.815	9.348	11.345
4	0.297	0.484	0.711	...	9.488	11.143	13.277

The Bold values signifies the calculation of 95 % confidence interval (refer equation 7 : $C < 7.815$) and 99 % confidence interval (refer equation 8 : $C < 11.345$)

A Chi-square distribution considers the terms ‘ Df ’ (Degrees of Freedom), which represent the number of unknowns. In our case, there are three unknowns, and, therefore, three degrees of freedom. Therefore, we can quickly obtain the probability that the above sum, and thus ‘ C ’ equals a particular value by using the Chi-square likelihood. As we are interested in a confidence interval, we are looking for the probability that ‘ C ’ is less than or equal to a particular value which can easily be obtained using the cumulative Chi-square distribution.

Using the Chi-square probabilities in Table 2 and Degree of Freedom = 3, we can find that,

$$P(C < 7.815) = 1 - 0.05 = 0.95 \quad (7)$$

And similarly,

$$P(C < 11.345) = 1 - 0.01 = 0.99 \quad (8)$$

From Eqs. (7) to (8) it is clear that the value for the constant ‘ C ’ will vary from 7.815 to 11.345. A 99 % confidence ellipse (red colored) is displayed along with the RGB distribution in Fig. 2. The confidence ellipse does not cover all the data, the reason being that the data are highly uncorrelated in nature. The primary cause of which is outliers. Outliers are values that are very low or very high as compared to the most values in the data set. Outliers should be removed before performing MANOVA.

4 Hypothesis generation

Multivariate analysis of variance, which is performed on the background model is used to generate the hypothesis that is evaluated for every differenced pixel obtained from the input image and the background model. The equation used is

$$\frac{x_R^2}{C.V_R} + \frac{x_G^2}{C.V_G} + \frac{x_B^2}{C.V_B} = 1 \quad (9)$$

where, x_R^2, x_G^2, x_B^2 are the RGB distribution of the differenced pixels. V_R, V_G, V_B are the variance obtained from the background model and constant ‘C’ varies from 7.815 to 11.345.

The hypothesis are stated as:

- NULL HYPOTHESIS: If the Eq. (9) satisfies, then the pixel belongs to the background and it is assigned the value ZERO.
- ALTERNATIVE HYPOTHESIS: If the null hypothesis is false, then pixel belongs to the foreground and it is assigned the value ONE.

But before testing the hypothesis on the differenced pixel, we have to remove the outliers as discussed in Sect. 3. We have several approaches of thresholding, such as Single threshold, multiple thresholds and adaptive thresholding discussed in the Sect. 1. As single threshold are not very useful in practical situations, therefore, two different thresholds are used in our algorithm. The only disadvantage is that the thresholds are generated through trial and error method for object extraction.

5 Experimental results and analysis

The proposed method with multivariate analysis on background variance for object detection has been evaluated on images with different illumination conditions, indoor and outdoor cases. Simulations are carried on image frames obtained from CAVIAR, MOT challenge benchmark and change detection datasets.

Cases with static background are taken from CAVIAR (Context Aware Vision using Image-based Active Recognition) project which includes people walking alone, fighting and passing out. The resolution is half-resolution PAL standard (384×288 pixels, 25 frames per second) and compressed using MPEG2. The file sizes are mostly between 6 and 12 MB, a few up to 21 MB. Change detection and MOT challenge dataset is used for cases exhibiting dynamic background motion.

The segmentation results are displayed in Figs. 3, 4, 5, 6, 7 and 8 with the following layout: (a) input image, (b) Averaged background model, (c) obtained result and (d) ground truth. Figures 3 and 7 are the cases for which the ground truths

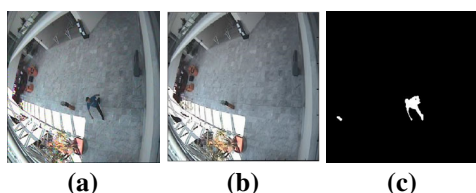


Fig. 3 Case 1: indoor scene with static background and low illumination

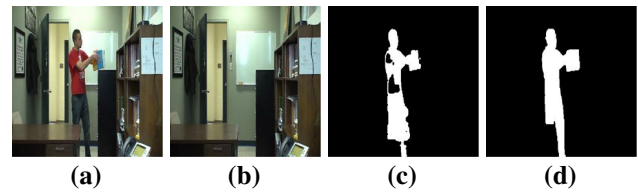


Fig. 4 Case 2: indoor scene with static background and moderate illumination

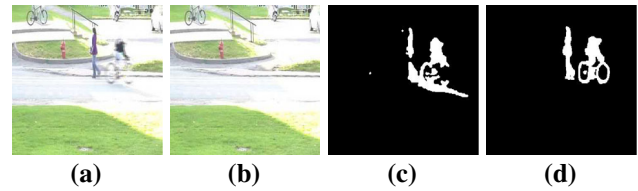


Fig. 5 Case 3: outdoor scene with static background and high illumination

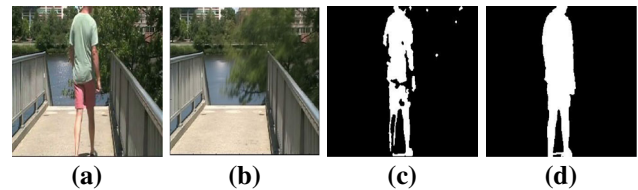


Fig. 6 Case 4: outdoor scene with dynamic background having water bodies, leaves and moving objects

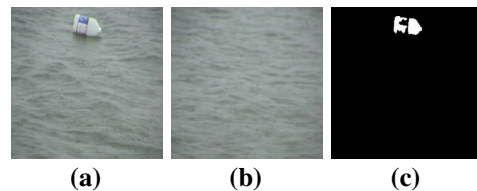


Fig. 7 Case 5: a floating object

are not available in the dataset, and, therefore, the performance evaluation of the method is done solely using obtained results. From the obtained results it is clear that the proposed method works well with indoor and outdoor scenes with the static and dynamic background having varying illumination. The only observed problem is the presence of cast shadow in high illumination cases (Case 3) and presence of false holes (Case 4) inside the objects silhouette. We can further process the output using morphological operations to reduce the effect of false holes.

5.1 Error rate

The error rate is obtained to evaluate the effectiveness of the algorithm. The error rate is given by the following equation:

$$\text{Error Rate} = \frac{\text{Error Pixel Count}}{\text{Frame Size}} \tag{10}$$

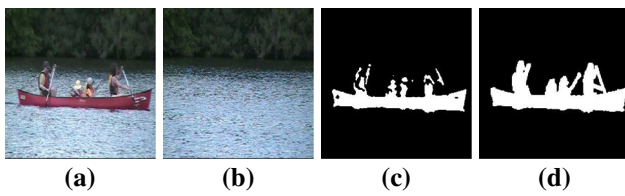


Fig. 8 Case 6: a canoe with a group of people

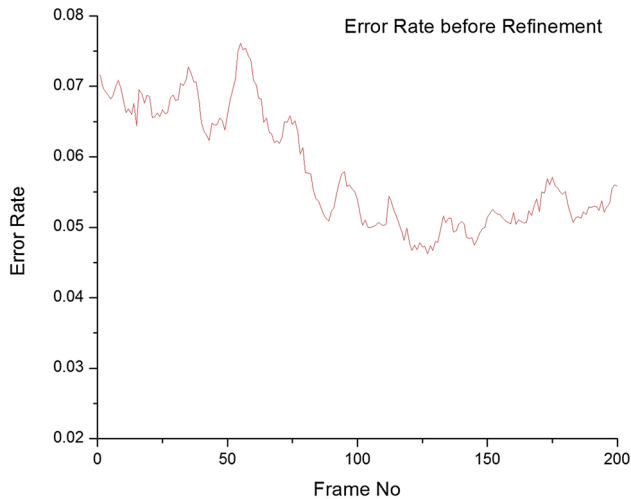


Fig. 9 Error rate in each frame for Case 4 before refinement

where error pixel count is the number of false positive and false negative pixels. Figures 9 and 10 shows the error rate for the Case 4. The error rate is minimized after refinement as shown in Fig. 10. In object detection, the error pixel count should be reduced as much as possible for accurate results. The accuracy of proposed algorithms is evaluated using ROC curve in Fig. 11, which is generated by fixing one of the thresholds and varying another. As depicted by the ROC curve the false positive, which stands for the number of background pixels detected as object pixels do not change over a large range, which is one of the plus points for the proposed algorithm.

5.2 Boundary displacement error

Boundary displacement error reflects the error in the obtained boundary and the actual boundary (ground truth boundary). The boundary of the result is achieved by first removing the holes inside the object using morphological operations and then by applying Canny edge detection. Ground truth boundary is obtained just by using Canny edge detection. The boundary in white color is the obtained boundary and red-colored is the actual boundary. The displacement error for four continuous frames is shown in Fig. 12. If the detected boundary pixels are exactly on the ground truth boundary then the error is zero; but if the pixel overlaps with a point on the dilated or eroded ground truth boundary with dilation

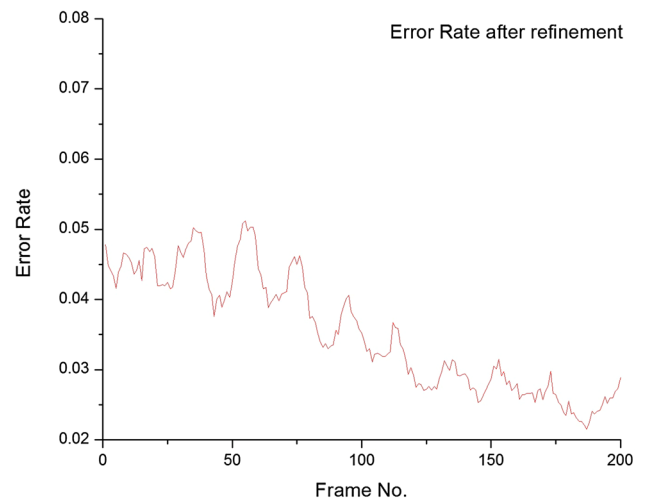


Fig. 10 Error rate in each frame for Case 4 after refinement

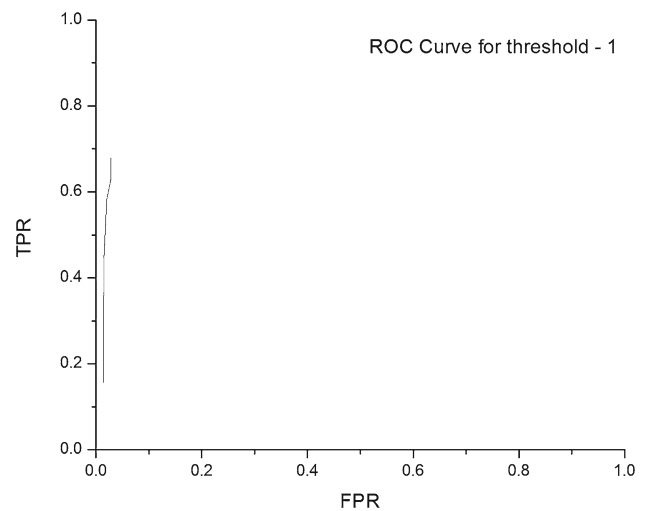


Fig. 11 ROC curve for object detection

or erosion radius ' r ', then the displacement error is of ' r ' or ' $-r$ ' pixels. From the output it is clear that we have either zero displacement error or negative error but no positive displacement error is obtained.

6 Shadow detection and removal

Segmentation of the moving object from its background is an important research topic in the recent past. But as shown in the experimental results (Figs. 4, 5, 6), the major issues with segmentation is cast shadows and self shadow. As a result, segmentation becomes inaccurate. Numerous shadow detection algorithms are available based on several color models. Major color models are RGB, HSV, HIS and YCbCr. In this section, we have discussed some of the Shadow Detection and Removal approaches from literature (Sects. 6.1, 6.2, 6.3)

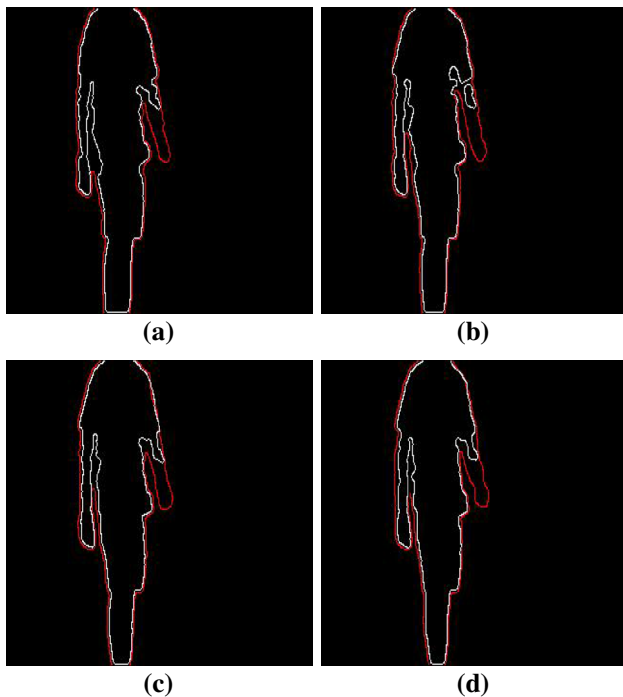


Fig. 12 Boundary displacement for four continuous frames

for better understanding of shadow removal and at last we proposed an automatic shadow removal approach for RGB color model in Sect. 6.4.

Shadow occurs when the object totally or partially occludes the light coming from the light source. Cast shadow can be defined as the darkened region on the background of an image that is due to the foreground objects blocking the light source, the presence of cast shadow can modify the perceived object shape. Self shadow is the part of the object that is not illuminated by direct light, the presence of self shadow modifies the perceived object shape and its color. The shadow has two parts to it, called umbra and penumbra. The umbra corresponds to the area where the direct light is almost totally blocked, whereas the area where light is partially blocked is called penumbra.

6.1 Shadow identification and classification using luminance and chrominance edge map

This method [30] propose to exploit color information for shadow detection by using the invariance properties of some color transformations. Among the traditional color features, normalized RGB, Hue (H) and saturation (S) are invariant features to shadows and shading. In addition to these well-known color spaces, new invariant color models, $c_1c_2c_3$ and $l_1l_2l_3$ are proposed in [31].

Optimum results were obtained using $c_1c_2c_3$ color model, and are defined as:

$$c_1 = \arctan \left(\frac{R}{\max(G, B)} \right) \tag{11}$$

$$c_2 = \arctan \left(\frac{G}{\max(R, B)} \right) \tag{12}$$

$$c_3 = \arctan \left(\frac{B}{\max(R, G)} \right) \tag{13}$$

The first step is to convert the input image in a color model sensitive to shadow and obtain an edge map using Sobel operator on the luminance component of the input image. Morphological operations can be applied if the edge map do not form the closed contours. The edge map helps in searching the shadow pixels in the portion of the image that is occupied by the object and its cast shadow. Second step is shadow classification in which the shadow pixels identified in the previous step is classified into cast and self shadow. Here a color edge map is obtained by logical OR operation on the edge map obtained with the Sobel operator on each color component. The color edge map detects shadow points which are occupied by the objects i.e., the self shadow and the remaining shadow pixels that were obtained form step 1st excluding self shadow pixels from step 2nd are classified as cast shadow. The flowchart for this method is given in Fig. 13.

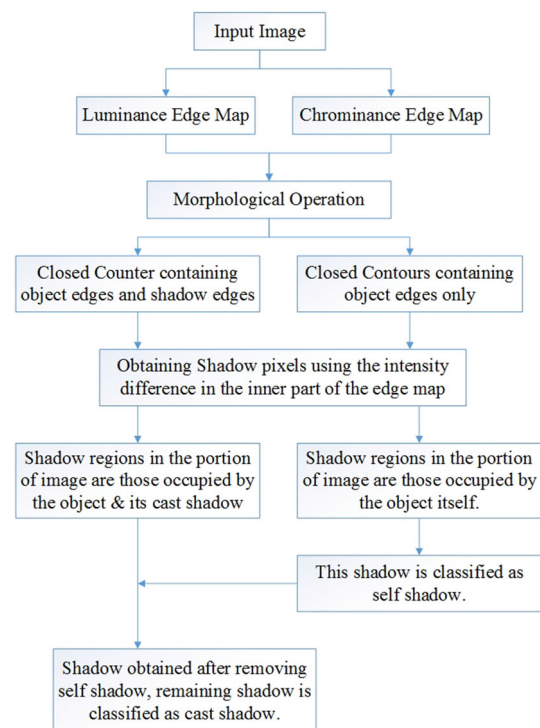


Fig. 13 Shadow identification and classification using two edge maps

6.2 Shadow detection using local and spatial information (statistical parametric approach)

There are several other ways to identify or to detect object and shadows. One such method is to obtain information from local, spatial or temporal domain. Local information is obtained based on the appearance of the individual pixels (a point in shadow gets darker compared to its appearance when illuminated). Spatial information is obtained from the neighborhood pixels as object and shadow inhibit compact region in an image. Temporal information are obtained from the relation of current frame and the previous frame.

The Statistical parametric approach of shadow detection [32], which was developed for ATON project make use of local information and can further combine it with spatial information. The flowchart of this approach is given in Fig. 14. This approach uses the concept that the probability density function of a shadowed pixel can be computed using change in the appearance of the pixels when shadowed, given its appearance when illuminated. An approximated linear transformation [33,34] is:

$$\vec{V} = D.V \quad \text{where, } V = [RGB]^T \tag{14}$$

is used to obtain the change in appearance. The diagonal matrix D is obtained as the slopes of the line fitted to plots between the shadow and background points for the three color components. Given the mean and variance of the color channels of the reference point, we can determine the mean and variance of pixels under shadow.

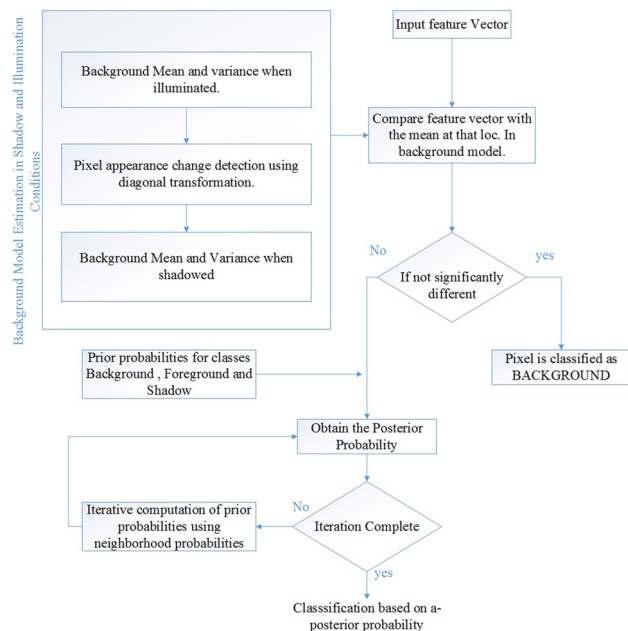


Fig. 14 Shadow identification using statistical parametric approach

Given, $(\mu_{IL}^R, \mu_{IL}^G, \mu_{IL}^B, \sigma_{IL}^R, \sigma_{IL}^G, \sigma_{IL}^B)$, the mean and variance of reference point and $D = \text{diag}(d_R, d_G, d_B)$, the diagonal matrix. we have,

$$\mu_{SH}^i = \mu_{IL}^i \cdot d_i \tag{15}$$

$$\sigma_{SH}^i = \sigma_{IL}^i \cdot d_i, i \in R, G, B \tag{16}$$

where, IL and SH represents illuminated and shadow.

Pixel segmentation is performed by estimating the a-posteriori probabilities separately for background, foreground and shadow classes. A pixel is then classified to the class having maximum a-posteriori probability.

$$p(C_i/v) = \frac{p(v/C_i)p(C_i)}{\sum_{j=1,2,3} p(v/C_j)p(C_j)} \tag{17}$$

where, v is the feature vector for a given pixel, $p(C_i)$ is the prior probability of i th class and $C_1 = \text{Background}$, $C_2 = \text{Shadow}$ and $C_3 = \text{Foreground}$.

Spatial constraints can further be imposed with the local information by updating the class membership probability based on the result of the neighboring pixels, which is then used to obtain new a-posterior probabilities for all the pixels.

6.3 Shadow detection using temporal information

This approach [35] is based on the idea that the shadow points can be detected as the points that are static for a short temporal sequence and are characterized by a constant luminosity change with respect to the reference background image. The first step is temporal image analysis, where two successive images I_{t-1} and I_t are compared to detect static points. Static point detection method uses radiometric similarity method to determine the similarity between two points.

$$R(p_i, q_i) = \frac{m[W_1(p_i)W_2(q_i)] - m[W_1(p_i)]m[W_2(q_i)]}{\sqrt{v[W_1(p_i)]v[W_2(q_i)]}} \tag{18}$$

where, m and v are the mean and variance estimated into small window (W_1, W_2) . Two points p_i and q_i are said to be static if their radiometric similarity is greater than 0.9.

The shadow points are stationary points s_i in the temporal sequence (between current image frame I_t and previous image frame I_{t-1}) and with respect to the corresponding background reference points b_i , differ by a constant factor A due to the change of luminosity (photometric gain), i.e.

$$A = \alpha_i = s_i/b_i. \tag{19}$$

Among all the static points recovered, the shadow points selected are those with a photometric gain less than 0.9. An iterative relaxation labeling method is used to remove all

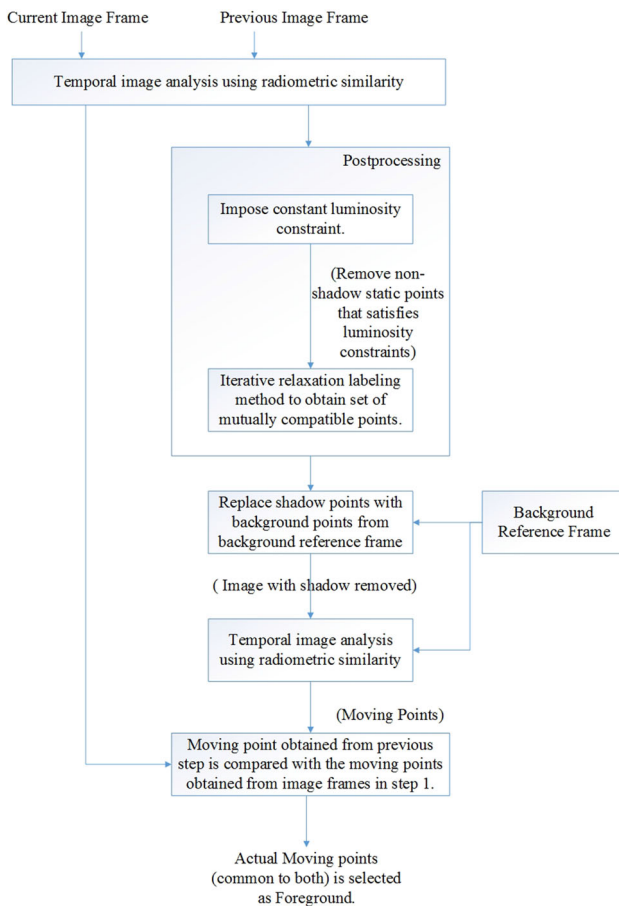


Fig. 15 Shadow identification using temporal information’s

those points that are not the part of static shadow but satisfies the constant luminosity gain. This algorithm searches for all those neighboring points that are mutually compatible according to the constraint. The mutually compatible points are selected as the final static points those having optimal photometric gain.

Next step in the process is to remove static shadow points with the points from the background reference frame to obtain the image with removed static shadow points. Temporal image analysis is again performed between background reference frame and frame with shadow removed, to obtain moving points. The moving points obtained are further compared with the previously obtained moving points between I_t and I_{t-1} . The points that are common to both are obtained as foreground points. Complete flowchart of the process is given in Fig. 15.

6.4 Automatic shadow removal using texture, luminance and chrominance differences

Cast shadow is the darkened region on the background of an image that is due to the foreground objects blocking the light source.

Considering the textural, luminance and chrominance properties or values of background and shadow pixels. The luminance values of the cast shadow pixels are normally lower than similar background pixels, whereas the chrominance values of the cast shadow pixels are similar to similar background pixels. And in terms of the textural property, the textural feature of a cast shadow is also very similar to background pixels. In other words, cast shadow does not alter the difference in the textural properties of the background and foreground pixels.

As texture based segmentation method [36] make use of the differences in textural property between the background pixels, shadow pixels and the object pixels itself, rather than just the intensity differences between them. Therefore, it is better to use all the three properties (textural difference, luminance difference and chrominance difference) for object segmentation. The proposed method for automatic shadow removal [37], considers all the three differences and then merging the output using logical OR operation to remove shadow.

This method comprises of two major steps.

- First step, is to calculate texture, luminance and chrominance difference (T_{diff} , L_{diff} , C_{diff}) and
- In step second, a threshold value is estimated from the histograms of these differences and finally TT_{diff} , TL_{diff} , TC_{diff} are computed by isodata thresholding method. An OR_{map} is then constructed by performing a logical OR operation of these thresholded differences.

Texture description of an image block is commonly calculated using the following autocorrelation function R ,

$$R(u, v) = \frac{(2M + 1)(2N + 1)}{(2M + 1 - u)(2N + 1 - v)} \times \frac{\sum_{m=0}^{2M-u} \sum_{n=0}^{2N-v} p(m, n)p(m + u, n + v)}{\sum_{m=0}^{2M} \sum_{n=0}^{2N} p^2(m, n)} \quad 0 \leq u \leq 2M, 0 \leq v \leq 2N \quad (20)$$

where u, v are the position displacements in the m, n direction, $2M + 1, 2N + 1$ are the dimensions of the image block, and $p(m, n)$ represents the intensity value at (m, n) .

The texture difference T_{diff} between two image blocks is calculated by mean square difference of two autocorrelation functions R to compare their similarities, where R_i, R_j are the autocorrelation functions R of two different image blocks.

$$T_{diff} = \frac{1}{(2M + 1)(2N + 1)} \sum_{u=0}^{2M} \sum_{v=0}^{2N} [R_i(u, v) - R_j(u, v)]^2 \quad (21)$$

The color model YCbCr is used to separate the luminance and chrominance components of the images to calculate L_{diff} and C_{diff} . A luminance difference L_{diff} between the input frame f_i and the background reference frame f_b is computed according to the following equation,

$$L_{diff} = \begin{cases} Y_b(x, y) - Y_i(x, y) & \text{If } Y_b(x, y) - Y_i(x, y) > 0 \\ 0 & \text{Otherwise} \end{cases} \quad (22)$$

Chrominance difference C_{diff} between input frame f_i and background reference frame f_b is computed using both Cb and Cr components according to the following equation,

$$C_{diff} = [Cb_i(x, y) - Cb_b(x, y)]^2 + [Cr_i(x, y) - Cr_b(x, y)]^2 \quad (23)$$

6.4.1 Experimental results

Complete flowchart for this approach is given in Figs. 16 and 17 is used to demonstrate the proposed algorithm.

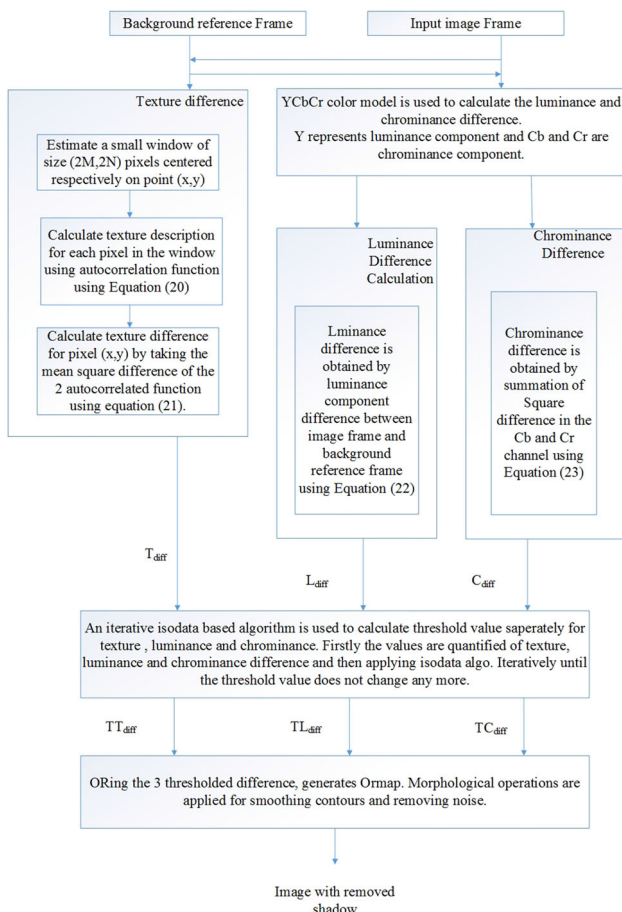


Fig. 16 Shadow identification using texture, luminance and chrominance information's

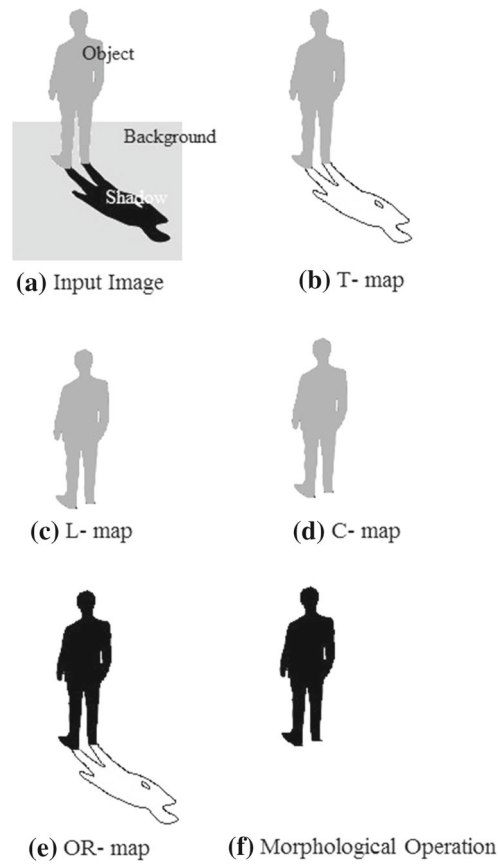


Fig. 17 Illustration of the proposed method. T-map is obtained by applying textural autocorrelation on input image. L-map and C-map are obtained through Luminance and Chrominance difference. OR-map is the logical-OR of **b–d**. Morphological operations are used to remove the shadow part to obtain final silhouette



Fig. 18 Background and object frames from four different videos

The proposed approach is tested on different real life samples extracted form various videos. Fig. 18a–d contains background reference frames used whereas Fig. 18e–h are the frames having moving objects. The result of silhouette extraction using automatic shadow removal method is depicted in Fig. 19. First row with Fig. 19a–d are the TT_{diff} ,

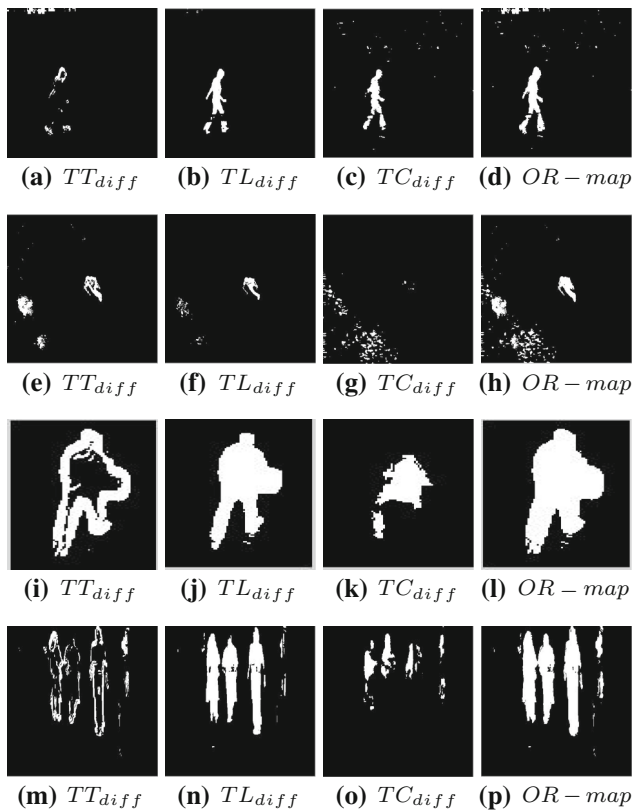


Fig. 19 Silhouette extracted using automatic shadow removal approach. TT_{diff} is the thresholded texture difference, TL_{diff} is the thresholded luminance difference and TC_{diff} is the thresholded chrominance difference, OR_{map} is the result of logical OR of the previous three results

TL_{diff} , TC_{diff} and OR_{map} outputs of first image frame in Fig. 18e and similarly, Fig. 19e–h are outputs of frame in Fig. 18f and so on. It can be easily inferred from the outputs generated that it works very well on shadow and there is no need of shadow removal as preprocessing step of silhouette extraction.

7 Conclusion

In this paper, we present MANOVA based foreground object extraction method with multiple thresholding. The threshold values are learned through experiments. This process can further be improved using adaptive thresholds. Cast shadow and false holes were the areas that need extra effort to remove. This method can work with the static or dynamic background and with varying degree of illumination as the error rate incurred is always less as false positive values remain almost constant and do not vary over the large range as compared to true positive values.

Several shadow detection and removal methods are also discussed in detail and an automatic shadow removal method

using texture, luminance, and chrominance is explained with results on different image frames. Experimental results show that silhouette extraction using statistical parametric, edge map and temporal approaches needs a pre-processing step of shadow removal while the proposed silhouette extraction method removes shadow by applying texture, luminance and chrominance differences as an inherent step and no pre or post processing is required. The noise obtained in the silhouette extraction results can be removed using further filtering or using morphological operations like erosion and pruning.

References

1. Mech R, Wollborn M (1998) A noise robust method for 2D shape estimation of moving objects in video sequences considering a moving camera. *Signal Process* 66:203217
2. Nicolas H, Labit C (1991) Global motion identification for image sequence analysis and coding. In: *Proceedings of speech and signal processing, IEEE international conference on acoustics*, pp 2825–2828
3. Smolic A, Sikora T, Ohm J-R (1999) Long-term global motion estimation and its application for sprite coding, content description and segmentation. *IEEE Trans Circ Syst Video Technol* 9:12271242
4. Aach T, Kaup A, Mester R (1993) Statistical model-based change detection in moving video. *Signal Process* 31:165180
5. Neri A, Colonnese S, Russo G, Talone P (1998) Automatic moving object and background separation. *Signal Process* 66:219232
6. Wren CR, Azarbayejani A, Darrell T, Pentland A (1997) Pfnder: realtime tracking of the human body. *IEEE Trans Pattern Anal Mach Intell* 19(7):780785
7. Stauffer C, Grimson WEL (2000) Learning patterns of activity using real-time tracking. *IEEE Trans Pattern Anal Mach Intell* 22(8):246252
8. Elgammal A, Duraiswami R, Harwood D, Davis LS (2002) Background and foreground modeling using nonparametric kernel density estimation for visual surveillance. *Proc IEEE* 90(7):11511163
9. Tsai DM, Lai SC (2009) Independent component analysis-based background subtraction for indoor surveillance. *IEEE Trans Image Process* 18(1):158167
10. Hsu YZ, Nagel H, Reckers G (1984) New likelihood test methods for change detection in image sequences. *Comput Vis Graph Image Process* 26(1):73106
11. Skifstad K, Jain R (1989) Illumination independent change detection for real world image sequences. *Comput Vis Graph Image Process* 46(3):387–399
12. Liu S, Fu C, Chang S (1998) Statistical change detection with moments under time-varying illumination. *IEEE Trans Image Process* 7(9):12581268
13. Li L, Leung MKH (2002) Integrating intensity and texture differences for robust change detection. *IEEE Trans Image Process* 11(2):105112
14. Haritaoglu I, Harwood D, Davis LS (2000) W4: real-time surveillance of people and their activities. *IEEE Trans Pattern Anal Mach Intell* 22(8):809–830
15. Amamoto N, Fujii A (1999) Detecting obstructions and tracking moving objects by image processing technique. *Electron Commun Jpn Part 3* 82(11):28–37
16. McKenna SJ, Jabri S, Duric Z, Rosenfeld A, Wechsler H (2000) Tracking groups of people. *Comput Vis Image Underst* 80(1):42–56

17. Seki M, Fujiwara H, Sumi K (2000) A robust background subtraction method for changing background. In: Proceedings of IEEE workshop applications of computer vision, pp 207–213
18. Ohta N (2001) A statistical approach to background suppression for surveillance systems. In: Proceedings of IEEE international conference on computer vision, pp 481–486
19. Koller D, Weber J, Huang T, Malik J, Ogasawara G, Rao B, Russel S (1994) Towards robust automatic traffic scene analysis in real-time. In: Proceedings of international conference on pattern recognition, pp 126–131
20. Cucchiara R, Grana C, Piccardi M, Prati A, Sirotti S (2001) Improving shadow suppression in moving object detection with HSV color information. In: Proceedings of international conference on intelligent transportation systems, pp 334–339
21. Heikkila M, Pietikainen M (2006) Texture based method for modeling the background and detecting moving object. *IEEE Trans Pattern Anal Mach Intell* 28(4):657–662
22. Otsu N (1979) A threshold selection method from gray-level histograms. *IEEE Trans Syst Man Cybern SMC*–9(1):62–66
23. Ridler TW, Calvard S (1978) Picture thresholding using an iterative selection method. *IEEE Trans Syst Man Cybern SMC*–8(8):630–632
24. Zack GW (1977) Automatic measurement of sister chromatid exchange frequency. *J Histochem Cytochem* 25(7):741753
25. Kita Y (2006) Change detection using joint intensity histogram. In: Proceedings of international conference on pattern recognition, Hong Kong, pp 351–356
26. Sen D, Pal SK (2009) Histogram thresholding using fuzzing and rough measures of associated error. *IEEE Trans Image Process* 18(4):879888
27. Bruzzone L, Prieto DF (2000) Automatic analysis of the difference image for unsupervised change detection. *IEEE Trans Geosci Remote Sens* 38(3):11711182
28. Rosin PL, Ellis T (1995) Image difference threshold strategies and shadow detection. In: Proceedings of Birmingham, UK, British machine vision conference, pp 347–356
29. Kapur JN, Sahoo PK, Wong AKC (1985) A new method for graylevel picture thresholding using the entropy of the histogram. *Comput Vis Graph Image Process* 29(3):273285
30. Salvador E, Cavallaro A, Ebrahimi T (2001) Shadow identification and classification using invariant color models. *IEEE Int Conf Acoust Speech Signal Process* 3:1545–1548
31. Gevers T, Smeulders AWM (1999) Color-based object recognition. *Pattern Recogn* 32:453–464
32. Mikic I, Cosman PC, Kogut GT, Trivedi MM (2000) Moving shadow and object detection in traffic scenes. *Proc Int Conf Pattern Recog*:321–324
33. Finlayson GD, Drew MS, Funt BV (1993) Diagonal transforms suffice for color constancy. *IEEE Int Conf Comput Vis*:164–171
34. Barnard K, Finlayson G, Funt B (1997) Color constancy for scenes with varying illumination. *Comput Vis Image Underst* 65(2):311–321
35. Branca A, Attolico G, Distanto A (2002) Cast shadow removing in foreground segmentation. 16th international conference on pattern recognition, vol 1, pp 214–217
36. Lam WWL, Pang CCC, Yung NHC (2003) A highly accurate texture-based vehicle segmentation method. *Opt Eng* 43(3):591603
37. Singh M, Nain N, Panwar S Foreground object extraction using thresholding with automatic shadow removal. 11th international conference on signal-image technology and internet-based systems (SITIS), pp 655–662