

Video Frame's Background Modeling: Reviewing the Techniques

Hamid Hassanpour, Mehdi Sedighi, Ali Reza Manashty

Department of Computer Engineering & IT, Shahrood University of Technology, Shahrood, Iran.
Email: h_hassanpour@yahoo.com, {sedighi_mahdi87, a.r.manashty}@gmail.com

Received January 20th, 2011; revised February 27th, 2011; accepted March 5th, 2011.

ABSTRACT

Background modeling is a technique for extracting moving objects in video frames. This technique can be used in machine vision applications, such as video frame compression and monitoring. To model the background in video frames, initially, a model of scene background is constructed, then the current frame is subtracted from the background. Eventually, the difference determines the moving objects. This paper evaluates a number of existing background modeling techniques in terms of accuracy, speed and memory requirement.

Keywords: Background Modeling, Moving Object

1. Introduction

Detection of objects or persons in a video sequence requires, in most of the techniques, that the background of the frame be omitted from the scene. A common method for extracting moving objects in video sequences is background subtraction [1,2]. This technique can be used in monitoring applications such as work place security, traffic control and video frame compression [3-5]. To detect the moving objects in video frames, initially, the model of scene background must be constructed (*i.e.* the image without the moving objects), then current frame is subtracted from the background model and eventually, the difference, determines the moving objects [6,7].

Background modeling can be classified into two main groups: non-statistical [8-10] and statistical approaches [2,11,12]. In the former group, the background image, usually from the initial frame, is modified along the frame sequences. In this approach, to extract the moving objects in the video sequences, the difference between the current frame and the background model is computed. Non-statistical approaches are fast, hence, they are suitable for real time applications.

The non-statistical background modeling presented in [1,13] namely RGABM, considers each pixel in a frame to be either as part of the moving object (simply the object) or the background. In this approach, the first frame is considered as the background and subsequent frames are subtracted from the background. Then the pixels with a value higher than a threshold are considered as the ob-

jects. In this approach, the background is updated along the frame sequences.

In the second group of background modeling approaches, the statistical based approaches, the probability distribution functions of the background pixels are estimated; then, in each video frame, the likelihood that a pixel belongs to the background is computed. The statistical based approaches have a better performance, compared to the non-statistical based approaches, in modeling background of the outer scenes. However, they may require more memory and processing time and hence be slower than the non-statistical based approaches.

One of the important statistical-based approaches to model the background image is the Gaussian mixture model. This approach uses mixture of models (multi-models) to represent the statistics of the pixels in the scene. The multimodal background modeling can be very useful in removing repetitive motion from, for examples shining water, leaves on a branch, or a wiggling flag [14, 15]. This approach is based on the finite mixture model in mathematics, and its parameters are assigned using the expectation maximization (EM) algorithm.

The non-parametric statistical-based background modeling presented in [11] can handle situations in which the background of the scene is cluttered and not completely static. In the other words, the background may have small wiggling motions, as it is in tree branches and bushes. This model estimates the probability of observing a specified value for a pixel in its previous values

obtained from older frame sequences. This model is frequently updated to adapt with the changes in the scene, hence to have a sensitive detection of moving targets.

There is a tradeoff between computational speed, memory requirement and accuracy in using the statistical based methods compared to non-statistical based methods. It is important for users to know the capabilities of different techniques, to choose the suitable method for their applications, which is the aim of this paper.

There are a number of issues need to be considered in any background modeling technique, they include detecting objects from the background, updating the background during time and extracting moving objects from video frames. These issues are considered as comparison factors in the evaluation process of this paper.

The rest of this paper is organized as following: Section 2, and Section 3, respectively, review a number of non-statistical and statistical background modeling methods. Experimental results of evaluating different background modeling methods on various videos are presented in Section 4. Finally, the paper is concluded in Section 5.

2. Non-Statistical-Based Background Modeling Methods

The non-statistical approaches suppose that the background is an image, usually from the initial frame, which is modified along the frame sequences. These approaches, aimed to extract the moving objects in the video sequences, use the difference between the current frame and the background model. The non-statistical methods are suitable for real time applications as they are considerably fast. To detect moving objects, in these approaches, subsequent frames are subtracted from the background, and then pixels with the value higher than a threshold are considered as the objects. A number of existing non-statistical background modeling methods are briefly described in the following subsections.

2.1. Background Modeling Independent of Time

This method is the simplest approach for computing background, which is independent of time; hence this method is named as BMIT (Background Modeling Independent of Time) [14,16,17]. In this approach, the first frame in video frame sequences is supposed to be the background and remains unchanged along the video sequences. The mathematical description of the background model can be represented as

$$B_{x,y}^k = I_{x,y}^0 \quad (1)$$

where $I_{x,y}^k$ is the pixel (x, y) of k -th captured frames, and $B_{x,y}^k$ is the pixel (x, y) of k -th background model.

2.2. The Improved Basic Background Modeling

BMIT suffers from noise and varying luminance in image sequence. The improved basic background modeling (IBBM) method was developed in [5] to alleviate the deficiencies of BMIT approach. Once the pixel value of the absolute difference frame is more than the threshold value, the pixel is regarded as part of the foreground; otherwise, it is assigned to the background. Whenever a pixel belongs to the moving object, it should be updated; otherwise, it is not essential to update. According to this idea, the mathematical description of IBBM can be expressed as the following [16]:

$$B_{x,y}^k = \begin{cases} I_{x,y}^k, & AD_{x,y}^k < T \\ B_{x,y}^{k-1}, & AD_{x,y}^k > T \end{cases} \quad (2)$$

where $AD_{x,y}^k$ is the pixel (x, y) of the absolute difference frame between the k -th captured frame and the $(k-1)$ -th background model, *i.e.*:

$$AD_{x,y}^k = |I_{x,y}^k - B_{x,y}^{k-1}| \quad (3)$$

The IBBM method can be used to reduce the noise effect and the varying luminance effect. However, IBBM has deckle effect hence it suffers from updating the deckle of the foreground in the background model [16,18]. On the other hand, if the foreground and background have similar colors then the wrong updating occurs.

2.3. The Long-Term Average Background Modeling

To solve the deckle effect problem of IBBM, the long-term average background modeling (LTABM) was suggested in [15,19] as defined below:

$$B_{x,y}^k = \frac{1}{k} \sum_{r=1}^k I_{x,y}^r \quad (4)$$

and its recursive model is as follows:

$$B_{x,y}^k = \left(1 - \frac{1}{k}\right) B_{x,y}^{k-1} + \frac{1}{k} I_{x,y}^k \quad (5)$$

The LTABM computes the average by involving the whole past frame up to the current frame. This approach depends on the number of frames (K). The smaller the number of frames, the larger the weight of each frame; hence, noise in each frame would be considered more. By increasing the number of frame, the weight of each frame is reduced; subsequently, the luminance variation would considerably generate amount of noise effect on the background.

2.4. The Moving Average Background Modeling

The moving average background modeling (MABM)

improves the LTABM, by employing the following definition for background model [10]:

$$B_{x,y}^k = \begin{cases} I_{x,y}^i, & -W+1 < i < 0 \\ \frac{1}{W} \sum_{r=k-(W-1)}^{k-1} I_{x,y}^r, & \text{others} \end{cases} \quad (6)$$

where W is the moving length. The background is the average of recent W captured frames. The weights of the last W frames are equal, but it cannot be written in a recursive form, which results in a very high memory requirement.

2.5. Running Gaussian Averaging Background Modeling

The Running Gaussian Averaging Background Modeling (RGABM) approach not only can be used to reduce the varying luminance effect and noise, but also can be written in a recursive form, as follows [16]:

$$B_{x,y}^k = \begin{cases} (1-\alpha)B_{x,y}^{k-1} + \alpha I_{x,y}^k, & k > 1, \alpha \in [0,1] \\ I_{x,y}^0, & k = 0 \end{cases} \quad (7)$$

where $B_{x,y}^{k-1}$ represents the background model in $k-1$ previous frames, and α is the background updating rate. In this method, the value of α is very important and is typically set to 0.05 [18].

3. Statistical-Based Background Modeling Methods

In statistical-based background modeling, the probability function of background is estimated. This function determines the probability for the belonging of the pixel to the background. Despite non-statistical based methods, these approaches are suitable for modeling outdoor and dynamic scenes. A number of these approaches are reviewed in the following subsections.

3.1. Gaussian Mixture Model

One of the challenging issues in background modeling is to model repetitive motions in the video such as the shining water, leaves of a branch, or a waving flag. *Stauffer and Grimson* in [2] have introduced the Gaussian mixture model (GMM) to extract the statistics of repetitive moving objects often exist in outdoor scenes.

This approach employs the finite mixture method [20] to estimate the background model. In finite mixture method, $f(x_i) = \{x_i | i=1, \dots, n\}$ can be estimated as the sum of c weighted kernels as below:

$$f(x) = \sum_{i=1}^c p_i g(x; \theta_i), \quad c \ll n \quad (8)$$

where p_i denotes the weight for the i -th kernel, $\sum_{i=1}^c p_i = 1$,

$p_i \geq 0$, and $g(x; \theta_i)$ is the probability density function with θ_i as the kernel density parameter. For each given feature vector x , it is considered as the background if $f(x) > \tau$.

In (8), the parameter should be set in a way that a higher value density function be assigned to the samples belong to the background. This issue causes the background and foreground pixels be classified with an acceptable accuracy.

The authors in [2] have used the EM as a valuable tool for optimizing problem. In using the EM technique, the number of kernel function must be given. In this approach, an initial estimate for the kernel parameters values is needed. After that, the parameters are updated following the new data value. The first step is to determine the posterior probabilities given by (9).

$$\hat{\tau}_{ij} = \frac{\hat{p}_i \phi(x_j; \hat{\mu}_i, \hat{\Sigma}_i)}{\hat{f}(x_j)} \quad i=1,2,\dots,c; j=1,2,\dots,n \quad (9)$$

where $\hat{\tau}_{ij}$ represents the estimated posterior probabilities that x_j belongs to the i -th kernel, $\phi(x_j; \hat{\mu}_i, \hat{\Sigma}_i)$ is the Normal density function of i -th kernel evaluated at x_j , $g(x, \theta_i)$ in (8) was assumed to be Normal with $\phi(x_j; \hat{\mu}_i, \hat{\Sigma}_i)$, and $\hat{f}(x_j)$ is the finite mixture estimated at x_j .

Indeed, the posterior probability in (9) provides the likelihood that a point belongs to each of the separate kernel densities. We can use this estimated posterior probability to obtain a weighted update of the parameters for each kernel. Following the EM algorithm, updated parameters for the mixing coefficients, the means, and the covariance matrices are obtained as bellow [20]:

$$w_i = \frac{1}{n} \sum_{j=1}^n \hat{\tau}_{ij} \quad (10)$$

$$\hat{\mu}_i = \frac{1}{n} \sum_{j=1}^n \frac{\hat{\tau}_{ij} x_j}{\hat{p}_i} \quad (11)$$

$$\hat{\Sigma}_i = \frac{1}{n} \sum_{j=1}^n \frac{\hat{\tau}_{ij} (x_j - \hat{\mu}_i)(x_j - \hat{\mu}_i)^T}{\hat{p}_i} \quad (12)$$

Stauffer and Grimson have used the following computationally simplified equations, as they can be implemented using recursive technique in programming [2]:

$$w_{k+1} = (1-\alpha)w_k + \alpha \cdot p(k|x_t, \theta_k) \quad (13)$$

$$\mu_{k+1} = (1-\rho_k)\mu_k + \rho_k x_t \quad (14)$$

$$\Sigma_{k+1} = (1-\rho_k)\Sigma_k + \rho_k (x_t - \mu_{k+1})(x_t - \mu_{k+1})^T \quad (15)$$

where

$$\alpha = \frac{1}{N+1} \quad (16)$$

$$\rho_k = \frac{\alpha \cdot p(k|x_t, \theta_k)}{w_k} \quad (17)$$

3.2. Non-Parametric Model

Previous methods assumed that the parameters of the model are unknown, and hence, they need to optimize the initial values of kernel parameters; but optimization is a time consuming operation. To overcome this problem, a non-parametric model was introduced in [1]. In this approach, there is no need to optimize the parameters of each kernel. For modeling the messy and fast wiggling behavior, the model must be updated continuously in order to capture the fast changes in the scene background.

For describing this model, let x_1, x_2, \dots, x_N be a recent sample of intensity values for a pixel. The probability density function, which indicates the pixel intensity value (x_t) at time t , can be estimated using the kernel estimator K as following:

$$\Pr(x_t) = \frac{1}{N} \sum_{i=1}^N K(x_t - x_i) \quad (18)$$

If we choose our kernel estimator function, K , to be a Normal function $N(0, \Sigma)$, where Σ represents the bandwidth of kernel function, then the density can be estimated using bellow equation [11]:

$$\Pr(x_t) = \frac{1}{N} \sum_{i=1}^N \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(x_t - x_i)^T \Sigma^{-1} (x_t - x_i)} \quad (19)$$

If we assume independency between the different color channels, and each color channel (j -th channel) has a different kernel bandwidth value of σ_j^2 , then the bandwidth matrix would be:

$$\Sigma = \begin{pmatrix} \sigma_1^2 & 0 & 0 \\ 0 & \sigma_2^2 & 0 \\ 0 & 0 & \sigma_3^2 \end{pmatrix} \quad (20)$$

and the density estimation is reduced to [20]:

$$\Pr(x_t) = \frac{1}{N} \sum_{i=1}^N \prod_{j=1}^d \frac{1}{\sqrt{2\pi\sigma_j^2}} e^{-\frac{1(x_{ij} - x_{ij})^2}{2\sigma_j^2}} \quad (21)$$

The pixel x_t is considered as part of foreground pixel if $\Pr(x_t) > t$ where the threshold t is a global threshold over the entire image that can be adjusted to achieve a desired accuracy.

Since we measure the deviations between two consecutive intensity values, the pair (x_i, x_j) usually comes from

the same local-in-time distribution and only few pairs are expected to come from cross distributions. If we assume that this local-in-time distribution is Normal (μ, σ^2) , then the deviation $(x_i - x_j)$ is has also a Normal distribution with $N(\mu, 2\sigma^2)$. Therefore, the standard deviation of the first distribution can be estimated as [11]:

$$\sigma = \frac{1}{0.68\sqrt{m}} \quad (22)$$

4. Performance Evaluation

In this study we have implemented the above described methods to evaluate their performance in terms of memory requirement, consuming time, and accuracy in background modeling. In the following sections, we discuss the above factors in each approach. Experimental results of the last two factors are also provided. In the evaluation process, the video frames are assumed to be gray-scale and the size of video frames is $(M*N)$ pixels.

4.1. Memory Requirement

In this section, we compare the memory requirements of the above described approaches: initially the statistical-based approaches, afterward the non-statistical based approaches.

4.1.1. Statistical Approaches

The GMM algorithm has two phases: the updating phase, and the moving objects extraction phase. In the updating phase, the mean, variance, and coefficient associated with each kernel (K) are updated. In addition, a number of features (d) are considered for each pixel. In the following discussions, the mean, variance, and coefficient are verified for a frame of video in using this approach.

The following assumptions are considered in evaluating the memory requirement:

1) For mean measurement:

- If dimension of input data is equal to d , then the mean vector dimension is also d for each pixel.
- A number of $M*N$ pixels are considered for each frame.
- The number of kernels is equal to K .

Consequently, the memory requirement order for mean measurement is equal to $O(K*M*N*d)$.

2) For covariance measurement:

- If the number of dimensions is d , then covariance matrix is a $d*d$ matrix.
- The number of kernels are K for each pixel, then $K*d*d$ arrays are required for each pixel.

According to above statements, covariance measurement memory requirement for a single frame is of $O(M*N*d*d*K)$.

3) For coefficient measurement:

- As the number of kernels for each pixel is K and each kernel has one coefficient, K coefficients are needed for each pixel.
- The number of total pixels in a frame is $M*N$.

According to the two aforementioned assumptions, the order of memory requirement in coefficient measurement is $O(M*N*K)$ for each frame.

Eventually, it is concluded that the first phase of GMM algorithm needs $O(M*N*K*d*d)$ units of memory.

The second phase of GMM algorithm extracts the moving objects. At this step, the values of mean, covariance, and coefficient for each pixel are stored in the memory. According to the first phase, the memory requirement for this step is also $O(M*N*K*d*d)$.

The second algorithm in statistical-based approaches is non-parametric background modeling. In this approach, memory requirement directly depends upon the kernel function. If the considered kernel function has a Normal distribution, the number of evaluating samples is equal to H , and the extracting feature dimension is equal to d for each pixel, then the required memory will have an order of $O(H*M*N*d*d)$.

4.1.2. Non-Statistical Approaches

The non-statistical approaches of background modeling consider the background as an image. All techniques in these approaches need $M*N$ units of memory for pixel-data storage except the MABM technique, which needs $L*M*N$ entries (L denotes the number of frames). **Table 1** summarizes the memory requirement for all of the techniques discussed above.

4.2. Time Consumption

In this section, we compute the time consumption of different background modeling techniques. As mentioned earlier, updating and extracting the moving objects are the two phases of the GMM algorithm. It is essential in

Table 1. Memory requirement for different background modeling approaches.

Approach Type	Algorithm Name	Memory Requirements
Statistical Approaches	GMM	$O(K*M*N*d*d)$
	Non-parametric background model	$O(K*M*N*d*d)$
	BMIT	$M*N$
Non-Statistical Approaches	IBBM	$M*N$
	LTABM	$M*N$
	MABM	$L*M*N$
	RGABM	$M*N$

the updating phase to compute (13)-(17) for each pixel. If data dimension is d , by assuming equal time consumption for multiplication and division operations, and neglecting the time consumption for addition and subtraction operations, then:

1) In (13), the order of time is $O(d*d)$.

2) In (14), the order of time is $O(d)$.

3) In (15), the order of time is $O(d*d)$.

4) In (16), the order of time is $O(1)$.

5) In (17), the order of time is $O(d*d)$.

6) If the model has k kernels, then order of consuming time for the updating phase would be $O(k*d*d)$ for each pixel, thus, order of time for each frame will be $O(M*N*k*d*d)$.

We need to use (19) to extract the moving object from the video sequences. In this equation, the order of time is $O(k*d*d)$ per pixel, therefore the time consumption of each frame will be $O(k*d*d*m*n)$.

In non-parametric method, by considering K and A as respectively the number of kernels and the time consumption to compute the belonging of a pixel to each kernel, the order of time for each pixel and each frame will respectively be $O(K*A)$ and $O(M*N*K*A)$. The order of time for BMIT is a constant, hence of order $O(1)$. For background modeling using IBBM, according to (3), the order of this function is $O(M*N)$. The LTABM method uses (15) for background modeling and this function contains two multiplications and one addition operation. Consequently, the order of time for LTABM will be of order $O(M*N)$. MABM algorithm, for constructing the model, calculates mean of L frames, hence the time consumption for this method is of order $O(L*M*N)$. Eventually the order of time for RGABM method, according to (9), is of order $O(M*N)$.

To intuitively evaluate the time consumption of different background modeling techniques, we applied them on a sample video containing 100 frames each with a dimension of 320×240 pixels. The experimental results have been provided in **Table 2**. The results demonstrate that the non-statistical methods are faster than statistical method; hence, these methods are suitable for real time applications. On the contrary, the consuming time of statistical methods (in particular, the non-parametric method) is much more than non-statistical methods; therefore, these methods cannot be used for real time applications. These results confirm the results in [5,10,16].

4.3. Accuracy

The last factor to be discussed in this paper is accuracy, which is the ability of the methods to discriminate objects from the background. For computing the accuracy, we define two error parameters. The first parameter is

Table 2. Time consumption of different techniques in modeling the background of a sample video containing 100 frames each with a dimension of 320×240 pixels.

Algorithm Type	Algorithm Name	Consuming Time	Order
Statistical approach	GMM	241.773229 s	$O(M*N*d*d)$
	Non-Parametric	981.430302	$O(M*N*H*A)$
Non-statistical approach	BMIT	9.817173 s	$O(1)$
	IBBM	10.200367 s	$O(M*N)$
	LTABM	10.253446 s	$O(M*N)$
	MABM	14.497409 s	$O(L*M*N)$
	RGABM	10.151088 s	$O(M*N)$

false negative (FN), the number of object pixels misclassified as background pixels over the total number of object pixels. The second parameter is false positive (FP), the number of background pixels misestimated as object pixels over the total number of background pixels. Noise or illumination changes are the causes of FP error.

We have used four frames from two video sequences, one indoor and one outdoor, to evaluate accuracy of each

method. The error rate parameters for the indoor video sequence are shown in **Table 3**. The FN results demonstrate that the lowest error rate belongs to MABM method; and the accuracy of GMM, LTABM, and RGAM methods are not acceptable. The high FN error rates express that parts of the objects are not extracted successfully, thus, extracted objects cannot be used for object indexing applications. The FP error rates demonstrate that the best result belongs to GMM and RGABM method, confirming that these methods are relatively safe against noisy conditions. The results from **Table 3** indicate that BMIT, IBBM, and MABM are more suitable for indoor background modeling compared to the other approaches evaluated in this research.

Table 4 represents accuracy of the approaches in modeling background of the outdoor video. The FN error rates of BMIT, IBBM and MABM methods are very high; therefore, the extracted objects using these approaches are incomplete. Consequently, these methods may not be suitable for real-world applications, which require completely identified objects. The results in **Table 4** indicate that the FN error rate in GMM and RGABM are less than 1%, however these methods have a high FP error rate. The results indicate that any of the evaluated approaches

Table 3. Comparing accuracy of different background modeling techniques for indoor applications.

Methods	Frames								Average	
	Frame 1		Frame 2		Frame 3		Frame 4		FN	FP
	FN	FP	FN	FP	FN	FP	FN	FP		
GMM	0	0.00037	0	0.0113	0.7548	0.0242	0.7337	0.0242	0.3721	0.0150
BMIT	0	0.0080	0	0.0253	0.1418	0.0739	0.2095	0.0684	0.0878	0.0439
IBBM	0	0.0077	0	0.0089	0.1442	0.0412	0.2267	0.0363	0.0927	0.0235
LTABM	0	0.00006	0	0.0295	0.6962	0.0267	0.8754	0.0434	0.3929	0.0249
MABM	0	0.0041	0	0.0194	0.1291	0.0615	0.2135	0.0781	0.0856	0.0407
RGABM	0	0.00023	0	0	0.7772	0.0183	0.8690	0.0123	0.4115	0.0077

Table 4. Comparing accuracy of different background modeling techniques for outdoor applications.

Methods	Frames								Average	
	Frame 1		Frame 2		Frame 3		Frame 4		FN	FP
	FN	FP	FN	FP	FN	FP	FN	FP		
GMM	0.0079	0.9748	0.0104	0.7911	0.0068	0.9416	0.0039	0	0.0073	0.6769
BMIT	0.1982	0.1275	0.2316	0.0950	0.2436	0.0664	0.1707	0	0.2110	0.0722
IBBM	0.1565	0.1619	0.1752	0.0984	0.1874	0.0701	0.1392	0	0.1646	0.0826
LTABM	0.0174	0.5693	0.0397	0.5921	0.0272	0.6922	0.0187	0	0.0258	0.4634
MABM	0.1422	0.1551	0.2133	0.1040	0.1860	0.1860	0.1248	0.1248	0.1666	0.1425
RGABM	0.0066	0.8187	0.0121	0.7225	0.0122	0.8265	0.0073	0	0.0096	0.5919

has either a high FN or a high FP, which is conceived that their parameters may need to be properly tuned for outdoor applications.

5. Conclusions

The background modeling methods can be classified into two main groups: statistical and non-statistical approaches. Approaches in the first group use statistical concepts for background modeling. Memory requirement and consuming time of these methods are very high; hence, statistical approaches are not suitable for real time applications. On the other hand, these methods can be suitable for outdoor environment if their parameters be properly tuned, because of their safe operation in noise and sudden change condition. However, non-statistical methods are very easy to implement, and memory requirement of these approaches is very low compared to statistical methods. In addition, time consumption of non-statistical methods is rather low; consequently, these methods are suitable for indoor environments and real-time applications.

REFERENCES

- [1] C. R. Wren, A. Azarbayejani, T. Darrell and A. P. Pentland, "Pfinder: Real-Time Tracking of the Human Body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, 1997, pp. 780-785. [doi:10.1109/34.598236](https://doi.org/10.1109/34.598236)
- [2] C. Stauffer and W. E. L. Grimson, "Adaptive Background Mixture Models for Real-Time Tracking," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 2, No. 23-25, 1999, pp. 246-252.
- [3] L. Hanzo, P. J. Cherriman and J. Streit, "Video Compression and Communications," WILEY, 2007. [doi:10.1002/9780470519929](https://doi.org/10.1002/9780470519929)
- [4] Y. Q. Shi, H. F. Sun, "Image and Video Compression for Multimedia Engineering," CRC Press, Boca Raton, 2008. [doi:10.1201/9781420007268](https://doi.org/10.1201/9781420007268)
- [5] P. Kumar, K. Sengupta and A. Lee. "A Comparative Study of Different Color Spaces for Foreground and Shadow Detection for Traffic Monitoring System," *Proceedings of the IEEE 5th International Conference on Intelligent Transportation Systems*, 2002, pp. 100-105.
- [6] N. Schofield and C. McFarlane, "Segmentation and Tracking of Piglets in Images," *Machine Vision and Applications*, 1995, pp. 187-193.
- [7] Cavallaro, A. Salvador and E. Ebrahimi, "Detecting Shadows in Image Sequences," *Visual Media Production (CVMP)*, 2004, pp. 165-174.
- [8] P. Blauensteiner, H. Wildenauer, A. Hanbury and M. Kampel, "On Colour Spaces for Change Detection and Shadow Suppression," *Proceedings of the 11th Computer Winter Vision Workshop (CVWWS6)*, 6-8 February 2006, pp. 117-123.
- [9] Erum A. Khan and E. Reinhard, "A Survey of Color Spaces for Shadow Identification," *APGV '04: Proceedings of the 1st Symposium on Applied perception in graphics and visualization*, New York, 2004, p.160.
- [10] M. Piccardi, "Background Subtraction Techniques: A Review," *IEEE International Conference on Systems, Man and Cybernetics*, Vol. 4, 2004, pp. 3099-3104.
- [11] A. M. Elgammal, D. Harwood and L. S. Davis, "Non-Parametric Model for Background Subtraction," *Proceedings of ECCV 2000*, 2000, pp. 751-767.
- [12] T. Horprasert, D. Harwood and L. S. Davis. "A Statistical Approach for Real-Time Robust Background Subtraction and Shadow Detection," *Proceedings of IEEE ICCV'99 FRAME-RATE Workshop*, 1999, pp. 1-19.
- [13] D. Koller, J. Weber, T. Huang, J. Malik, G. Ogasawara, B. Rao and S. Russell, "Towards Robust Automatic Traffic Scene Analysis in Real-Time," *Proceedings of the 12th IAPR International Conference on Computer Vision & Image Processing*, Vol. 1, 1994, pp. 126-131.
- [14] N. Martel-Brisson and A. Zaccarin, "Moving Cast Shadow Detection from a Gaussian Mixture Shadow Model," In *CVPR '05: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, Vol. 2, 2005, pp. 643-648.
- [15] A. Prati, I. Mikic, M. M. Trivedi and R. Cucchiara, "Detecting Moving Shadows: Algorithms and Evaluation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 25, No. 7, 2003, pp. 918-923. [doi:10.1109/TPAMI.2003.1206520](https://doi.org/10.1109/TPAMI.2003.1206520)
- [16] S.-T. Su and Y.-Y. Chen, "Moving Object Segmentation Using Improved Running Gaussian Average Background Model," *Digital Image Computing: Techniques and Applications*, 2006, pp. 24-30.
- [17] N. Oliver, B. Rosario and A. Pentland, "A Bayesian Computer Vision System for Modeling Human Interactions," *International Conference on Vision Systems*, Springer, 1999.
- [18] R. Jain, R. Kasturi and B. G. Schunk, "Machine Vision," McGraw-Hill Inc., 2003, pp. 63-69.
- [19] N. Friedman and S. Russell, "Image Segmentation in Video Sequences: A Probabilistic Approach," *Proceedings of the 13th Conference on University in Artificial Intelligence (UAI)*, 1997.
- [20] L. Wendy and A. R. Martinez, "Computational Statistics Handbook with Matlab," *CRC Press*, 2002.