LOBS: Local Background Subtracter for Video Surveillance

Kalyan Kumar Hati, Pankaj Kumar Sa, and Banshidhar Majhi
Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela – 769 008, Odisha, India
{KalyanKumarHati, PankajKSa, BMajhi}@nitrkl.ac.in

***Abstract*—In surveillance system video sequences are obtained through static cameras and fixed background. A popular approach called background subtraction is generally used in this scenario. Existing approaches in this field try to detect the object first and then remove the shadow in the subsequent phase. Here we have tried to combine both object detection and shadow removal module to a single module. In this work, a background model is proposed based upon the stationary pixels across the frames required for background model initialization. Considering the stationary and non-stationary pixel information background model is developed, which is used for background subtraction in subsequent phase. A local thresholding based background subtraction technique is proposed for foreground object extraction and removal of shadow. Experimental results shows that our method outperforms many state-of-the-art techniques. The proposed technique is robust to challenges like pose and illumination variations.**

***Index Terms*—Video Surveillance, Background Model, Background Subtraction, Segmentation, Object Detection.**

I. Introduction

Object detection and tracking in video is an important computer vision application that has been vividly researched in the past decades. It consists of two closely related processes, object detection and object tracking. Object detection involves locating an object in the frames of a video sequence. Shadow being an integral part of the real time situation is also detected in this process. The shadow detected are removed in subsequent phase before tracking of the detected object. Object tracking represents the process of monitoring the objects' spatial and temporal changes during the movie sequence in a video. Object tracking can be applied in many areas like automated surveillance, traffic monitoring, human computer interaction etc. Most of the surveillance system includes static cameras and fixed background, which gives a clue for the object detection in videos by background subtraction technique. The basic principle of background subtraction is to compare a static background frame with the current frame of the video scene pixel by pixel. This technique builds a model of the background and each successive frame is compared with the model to detect zones where a significant difference occurs. The purpose of a background subtraction algorithm is, therefore, to distinguish moving objects referred to as *foreground*, from static or slow moving parts of the scene called *background*. Moving objects in a scene can be obtained by comparing each frame of the video with a background model, where the background model is the representation of the scene. The above process is called as *background subtraction* [1].

The proposed technique determines the stationary and non-stationary pixels in the frames required for background modeling. This pixel information is then used to model the background. Any incoming frame is then compared with the prepared model to detect the foreground objects and remove the shadow.

The rest of the paper is organized as follows: Section II describes the related works. In section III the proposed algorithm is presented. Simulation results of proposed algorithm with different videos are presented in section IV. Further, to show the efficiency of the proposed scheme, time and space complexity analysis are shown in section IV-C. Finally, section V deals with the concluding remarks.

II. Related Works

A significant amount of work is contributed by many researchers in object detection and tracking. Challenges in this area constrained the researchers to work with many presumptions. The following paragraphs review some of them.

Filters were used for background modeling and subtraction long years back. One such method is described by Koller *et al.* in [2], which addresses the problem of multiple car tracking with occlusion reasoning. They have employed a contour tracker, based on intensity and motion of boundaries. In order to achieve this they have used linear Kalman filter in two ways, one for estimating the motion parameters and another for estimating the shape of the contour of the car. Maintenance of background model being an important aspects of background modeling and subtraction, Toyama *et al.* in [3] developed a three component system for background maintenance: the pixel level component performs Wiener filtering to make probabilistic predictions of the expected background, the region-level component fills in homogeneous regions of foreground objects, and the frame-level component detects sudden and global changes.

Wren *et al.* in [4] have proposed to model the background independently at each pixel location (i, j). The model is based on computation of Gaussian probability density function (pdf) on the last n pixels values. In order to avoid the pdf calculation from beginning at each new frame, time t, a running average is computed as follows,

$$\mu_t = \alpha I_t + (1-\alpha)\mu_{t-1} \tag{1}$$

where I_t is the pixel's current value, μ_t is the previous average, and α is an empirical weight. The other parameter of the Gaussian probability density function, the standard deviation σ_t, can be computed similarly. In addition to speed, the advantage of the running average is given by the low memory requirement for each pixel. Here each pixel consists of the two parameters (μ_t, σ_t) instead of the buffer with the last n pixels values. At each t frame time, the I_t, pixel's value can then be classified as a foreground pixel if the inequality in following equation holds;

$$|I_t - \mu_t| > k\sigma_t \tag{2}$$

otherwise, I_t will be classified as background. Koller *et al.* in [5] explained that, (1) is more often updated, hence they proposed to modify the model as:

$$\mu_t = M\mu_t + (1-M)(\alpha I_t + (1-\alpha)\mu_{t-1}) \tag{3}$$

where the binary value M is 1 in correspondence of a foreground value, and 0 otherwise.

Lo and Velastin in [6] proposed to use the median value of the last n frames as the background model. Cucchiara *et al.* in [7] corroborated that such a median value provides an adequate background model even though the n frames are subsampled with respect to the original frame rate by a factor of 10. The main disadvantage of a median-based approach is that its computation requires a memory with the recent pixels values.

Stauffer and Grimson in [8] developed a complex procedure to accommodate permanent changes in the background scene. The procedure is named as Mixture of Gaussian. Here each pixel is modelled separately by a mixture of K Gaussian

$$P(I_t) = \sum_{i=1}^{K} \omega_{i,t}\eta\left(I_t; \mu_{i,t}, \Sigma_{i,t}\right) \tag{4}$$

where K can take the value between 3 and 5.

Elgammal *et al.* in [9] proposed to model the background distribution by a non-parametric model based on Kernel Density Estimation (KDE) on the buffer of the last n background values. According to [10] KDE guarantees a smoothed, continuous version of the histogram. In [9] the background pdf is given as a sum of Gaussian kernels centred in the most recent n background values, x_i:

$$P(x_t) = \frac{1}{\mathrm{n}} \sum_{i=1}^{n} \left(x_t - x_i, \Sigma_t\right) \tag{5}$$

The method described by Seki *et al.* in [11] is based on the assumption, neighbouring blocks of background pixels should follow similar variations over time. While this assumption holds most of the time, especially for pixels belonging to the same background object, it becomes problematic for neighbouring pixels located at the border of multiple background objects.

Few samples are collected over time and used to train a principal component analysis (PCA) model. A block of a new video frame is classified as background if the observed image pattern is close to its reconstructions using PCA projection coefficients of eight-neighbouring blocks. Such a technique is also described by Power and Schoonees in [12], but it lacks an update mechanism to adapt the block models over time. Oliver *et al.* in [13] focused on the PCA reconstruction error. A similar approach, the independent component analysis (ICA) of serialized images from a training sequence, is described by Tsai and Lai in [14] for training of an ICA model. The resulting demixing vector is then computed and compared to that of a new image in order to separate the foreground from a reference background image. The method is said to be highly robust to indoor illumination changes.

A two-level mechanism based on a classifier was introduced by Lin *et al.* in [15]. This classifier first determines whether an image block belongs to the background. Appropriate block wise updates of the background image are then carried out in the second stage, depending upon the results of the classification. The scheme proposed by Maddalena and Petrosino [16] also works on the basis of classification, where the background model learns its motion patterns by self organization through artificial neural networks.

The W4 model presented by Haritaoglu *et al.* in [17] is a rather simple and effective method. It uses three values to represent each pixel in the background image: the minimum and maximum intensity values, and the maximum intensity difference between consecutive images of the training sequence. Gutchess *et al.* in [18] proposed a background model in which multiple hypotheses of the background value at each pixel were generated by locating periods of stable intensity in the sequence. The likelihood of each hypothesis is then evaluated using optical flow information from the neighbourhood around the pixel, and the most likely hypothesis is chosen to represent the background. Jacques *et al.* in [19] brought a small improvement to the W4 [17] model together with the incorporation of a technique for shadow detection and removal. C.R. Jung [20] proposed a new background subtraction algorithm with shadow identification. In the training stage, robust estimators are used to model the background, and a fast test is used to detect foreground pixels in the evaluation stage. A statistical model is combined with expected geometrical properties for shadow identification and removal. Finally, morphological operators are applied to remove isolated foreground pixels.

Barnich and Droogenbroeck [21] proposed a universal background subtraction algorithm called ViBe for video sequences. In ViBe, for each pixel a set of values taken in the past, at the same location or in the neighbourhood is stored. It then compares this set to the current pixel value in order to determine whether that pixel belongs to the background, and adapts the model by choosing randomly which values to substitute from the background model.

Kim and Kim [22] introduced a novel background subtraction algorithm for temporally dynamic texture scenes. The proposed algorithm adopt a clustering-based feature, called fuzzy color histogram (FCH), which has an ability of greatly attenuating color variations generated by background motions while still highlighting moving objects.

From the literature it is observed that simple methods reported are not so robust but in turn complex methods do so at a very high computational cost. Moreover they have tried to detect the object and remove the shadow from the object in two different modules. In view of the above, here we proposed a computationally efficient algorithm based on local thresholding which combines two stage processes of object detection and shadow removal to single a stage. Further, it is shown that the scheme is illumination and pose variant. Comparative analysis is performed with existing schemes.

III. Proposed Algorithm

The proposed technique, LOcal Background Subtracter (LOBS) consists of two stages. First stage of the method tries to find out the *stationary pixels* in the frames required for background modeling. In the subsequent part, a background model is developed by considering stationary pixels obtained above. In second stage, a local thresholding based background subtraction is described, which tries to find out the foreground object by comparing any frame of the video with background model obtained above. LOBS takes two constant parameters; a single dimensional window of size W (an odd length window) and a constant C for thresholding. The optimal value for these parameters are described in section IV.

A. Developing a Background Model

In order to develop a background model, first few frames of the video are considered. Let V be an array containing k consecutive frames, which are initially considered. $V^s(i,j)$ be the intensity of a pixel (i,j) in the s^{th} frame. A single dimensional window of size W (an odd length window) across the k consecutive frames is considered. Now the same is moved from 1^{st} frame to k^{th} frame of V. During each pass we have collected the $\lceil W \div 2 \rceil^{th}$ element and absolute deviation of each element from the same is found. A pixel is designated as stationary or non-stationary by looking to the deviation of central element with all other elements. This process is repeated for $h \times w$ times, where h and w represent the height and width of the frame respectively. At the end of this process, the stationary and non-stationary pixels are classified in $k - W - 1$ number of frames out of k frames in V for background modeling.

The initial background model $\mathbf{B}$ for a pixel (i,j) is represented by a vector as —

$$\mathbf{B}(i,j) = \begin{bmatrix} m(i,j) \\ n(i,j) \end{bmatrix} = \begin{bmatrix} \min\limits_z V_z(i,j) \\ \max\limits_z V_z(i,j) \end{bmatrix} \quad (6)$$

where $V_z(i,j)$ are the stationary pixels of the z^{th} frame.

B. Extraction of Foreground Object

Pixel of any input frame $I_t(i,j)$ of the video sequence is compared with $\mathbf{B}(i,j)$. A pixel is classified as background when the following equality holds —

$$\{m(i,j) - \epsilon(i,j)\} \ \leq \ I_t(i,j) \ \leq \ \{n(i,j) \ + \epsilon(i,j)\} \quad (7)$$

where the threshold $\epsilon(i,j)$ is computed as —

$$\epsilon(i,j) = \frac{1}{C}\left[m(i,j) + n(i,j)\right] \quad (8)$$

The frame obtained here is called as segmented frame containing foreground object with shadow suppressed. It has been found that some of the background pixels are misclassified as foreground pixel. Those pixels are suppressed through morphological post processing.

This concludes the description of our algorithm LOBS. Since our algorithm successfully detects object and removes the shadow according to local illumination in the scene, hence we name it as LOcal Background Subtracter (LOBS).

IV. Results and Discussion

In this section we have determined the optimal values for the parameters of LOBS. Thereafter simulation results of LOBS are presented. Finally, an asymptotic comparison of LOBS with seven other state-of-the-art techniques is presented.

For the sake of comparison, we have produced manually ground-truth segmentation maps for subsets of frames taken from three test sequences. The first sequence (called “shadow indoor”) is an indoor sequence recorded in a large hall. This video is recorded with only one fluorescent lamp switched on, which is not sufficient enough to light the entire hall and thereby ensuring the illumination invariance. The second sequence (“hall monitor”) was obtained from Center for Image Processing Research (CIPR) unit of Rensselaer Polytechnic Institute, New York, USA. The third and final sequence (called “shadow outdoor”) is an outdoor video sequence, where the varying illumination environment is ensured by recording it in a partially cloudy day. The “shadow outdoor” sequence is used below to determine optimal value for parameters of LOBS.

Different metrics can be used to evaluate the output of a background subtraction algorithm given a series of ground-truth segmentation maps. These metrics usually involve the following quantities: the number of true positives (TP), which counts the number of correctly detected foreground pixels and the number of true negative (TN), which counts the number of correctly detected background pixels. Sum of TP and TN will give the number of correctly classified pixel in the frame. So, the percentage of correct classification (PCC), can be calculated as,

$$PCC = \frac{TP + TN}{Total\ number\ of\ pixels\ in\ the\ frame} \quad (9)$$

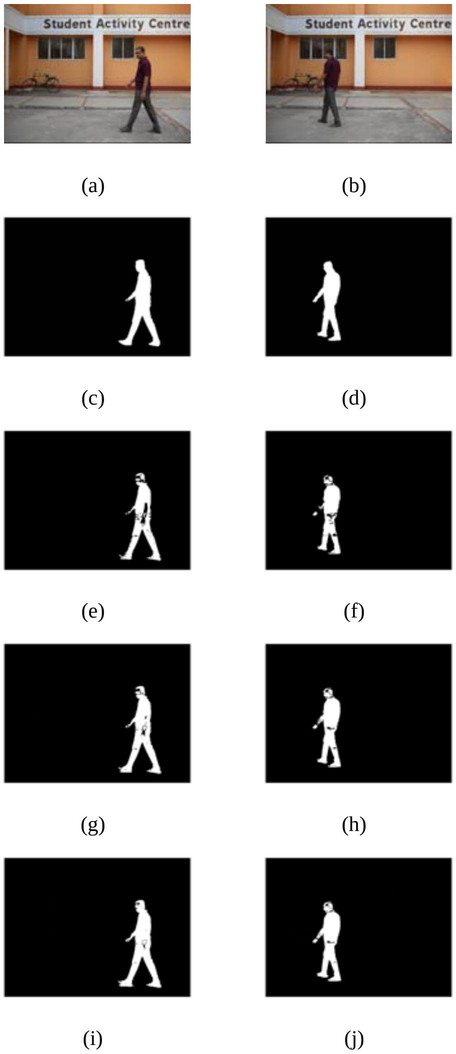

Fig. 1. Results of “shadow outdoor” sequence to determine optimal value of C using $W = 9$. (a) $frame$ 78, (b) $frame$ 136, (c) ground truth of $frame$ 78, (d) ground truth of $frame$ 136, (e) segmented frame at $C = 5$ of $frame$ 78 (98.55), (f) segmented frame at $C = 5$ of $frame$ 136 (98.43), (g) segmented frame at $C = 7$ of $frame$ 78 (99.45), (h) segmented frame at $C = 7$ of $frame$ 136 (99.43), (i) segmented frame at $C = 10$ of $frame$ 78 (99.19), (j) segmented frame at $C = 10$ of $frame$ 136 (99.07).

A. Determination of LOBS Parameters

The proposed model LOBS has two parameters:

- the single dimensional window size W to classify a pixel as stationary or non-stationary;
- a constant C for local thresholding to detect objects and remove shadow;

Higher order window size takes more computation and lower order provides poor result of W. Therefore, the value of W is chosen as 9 which is an educated choice. Keeping the value of $W = 9$, we have tested for various values of C and calculated the PCC according to equation 9. Different values considered for C are $5, 6, 7, 8, 9, 10$. Some of the simulation results are shown in Fig 1. The numerical value in the bracket, in caption of figures indicates the PCC value.

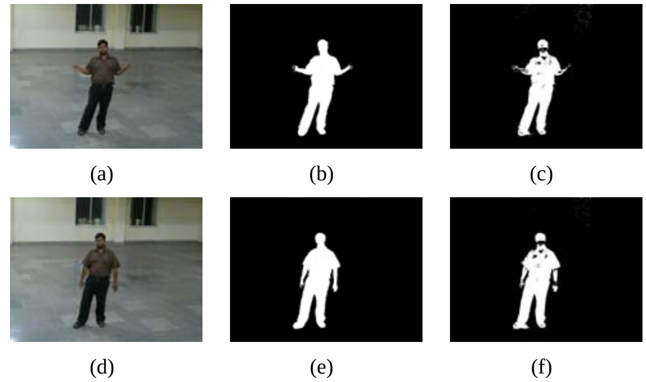

Fig. 2. Results of “shadow indoor” video sequences. (a) original frame ($frame$ 135), (b) ground-truth, (c) LOBS (99.41), (d) original frame ($frame$ 182), (e) ground-truth, (f) LOBS (99.43).

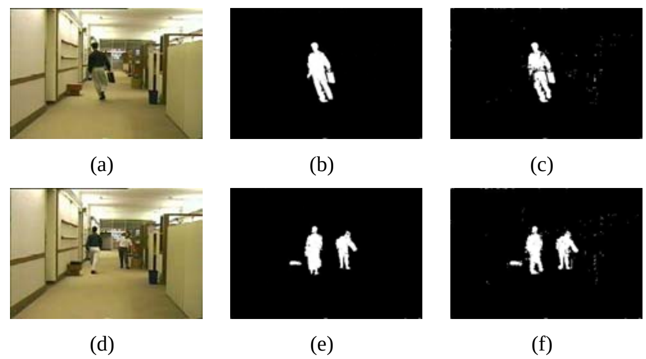

Fig. 3. Results of “hall monitor” video sequences. (a) original frame ($frame$ 83), (b) ground-truth, (c) LOBS (99.39), (d) original frame ($frame$ 178), (e) ground-truth, (f) LOBS (99.42).

B. Simulation Results

Here we present the simulation results of LOBS. As per the discussion in earlier part of this section we have considered two frames from each of the three sequences. Figure 2 to Figure 4 describes the simulation results. The numerical value in the bracket, in caption of figures indicates the PCC value.

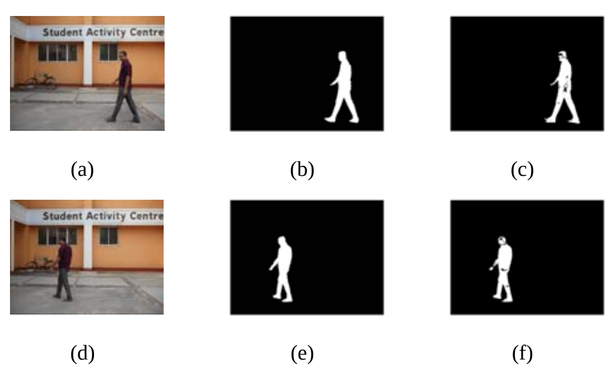

Fig. 4. Results of “shadow outdoor” video sequences. (a) original frame ($frame$ 78), (b) ground-truth, (c) LOBS (99.41), (d) original frame ($frame$ 136), (e) ground-truth, (f) LOBS (99.37).

C. Asymptotic Analysis

An asymptotic comparison of LOBS with seven other state-of-the-art techniques is described here. In the later part of this paragraph a detailed comparison between background modeling and subtraction of W4 algorithm [17] and LOBS is presented.

Among the methods reviewed, the temporal median filters [6] and [7] are the fastest, where for each pixel, the classification is just a threshold difference between the background model and frames. Background model updation in such scenario can be approximated as linear in the number of samples n_s (n_s is sub-sampled from the sample set n, where n is the total number of frames in the video). The corresponding complexity can be stated as $O(n_s)$. The Mixture of Gaussian [8] method has $O(m)$ complexity, where m is the number of Gaussian distributions used, typically in the order of 3–5. In order to classify a new pixel, the KDE model as described in [10] and [9] computes its value in the Gaussian kernels centred on the past n frames, thus raising $O(n)$ complexity, with n typically as high as 100. According to authors of [17] and [19] time complexity is found to be $O(n_s)$. Looking to the proposed algorithm the time complexity similarly can be found as $O(n_s)$. In most of the reviewed methods it has been found that the space complexity is same as time complexity. The algorithm proposed here also has same result. Table I gives a summary of the results.

TABLE I
ASYMPTOTIC COMPARATIVE ANALYSIS

Method	Time Complexity	Space Complexity
Temporal median filter [6], [7]	$O(n_s)$	$O(n_s)$
Mixture of Gaussian [8]	$O(m)$	$O(m)$
Kernel density estimation [10], [9]	$O(n)$	$O(n)$
W4 [17] and Jacques Jr.*et al.* [19]	$O(n_s)$	$O(n_s)$
LOBS	$O(n_s)$	$O(n_s)$

Contributions of the authors in W4 [17] is towards activities recognition of people in real-time surveillance. Of the total system proposed, we have compared our algorithm with that of background modeling and subtraction part of W4 in the following paragraphs.

Time complexity of LOBS to find the stationary pixels is $O(hw(n_s - 8)W \log W)$, where W refers to window size, which is 9. The height and width of the frame is represented by h and w respectively. In the case of W4 the time complexity of finding the stationary pixels is $O(hwn_s \log n_s)$. From the space complexity point of view, proposed one uses a single dimensional window of size 9 and W4 uses two variables to accommodate the mean and standard deviation. So, the space complexity is found to be $O(9)$ in our case and and $O(2)$ in W4. In order to construct the background model, LOBS uses a two dimensional model as compared to the three dimensional model of W4. Time complexity for background modeling of LOBS and W4 are $O(hw(n_s - 8)\log(n_s - 8))$ and $O(hwn_s \log n_s)$ respectively. From the space point of view, we have $O(2hw)$ as compared to $O(3hw)$ of W4. This shows an improvement in space requirement for modeling the background in LOBS.

The time complexity for foreground object extraction (Subsection III-B) is $O(hw)$ and is same for W4 as well. Space complexity of the proposed algorithm is $O(1)$ and that of the W4 is $O(hw + 1)$. From here it can also be concluded that even though the time complexities for both the algorithms are same, LOBS has an edge over W4 in space complexity.

Neglecting n_s(as it very small in the range 20 to 30) and constant, the above discussion can be summarized as follows:

- Time complexity of both the algorithm is found to be $O(n_s hw)$. So, this indicate that LOBS is asymptotically as efficient as W4.
- Space complexity of LOBS is found to be $O(2hw)$ as against $O(4hw)$ of W4. So, this indicate that LOBS is twice better than compared algorithm.

V. CONCLUSION

Videos obtained from surveillance system contains fixed cameras and static background. In such scenario object detection can be achieved with background subtraction technique. In this article a background model is proposed based on stationarity of the pixel. The background subtraction technique based on local thresholding successfully segments the objects to be tracked and successfully eliminates the shadow pixels. Simulation results shows that our method outperforms many state-of-the-art techniques. LOBS is robust to illumination and pose.

REFERENCES

[1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Computing Survey*, vol. 38, Dec 2006.

[2] D. Koller, J. Weber, and J. Malik, "Robust multiple car tracking with occlusion reasoning," in *Proc. of European Conf. on Computer Vision*. Springer-Verlag, May 1994, pp. 189–196.

[3] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers, "Wallflower: principles and practice of background maintenance," in *Proc. of the Seventh IEEE Int. Conf. on Computer Vision*, vol. 1, Sep 1999, pp. 255–261.

[4] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, "Pfinder: real-time tracking of the human body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 780–785, Jul 1997.

[5] D. Koller, J. Weber, T. Huang, J. Malik, G. Ogasawara, B. Rao, and S. Russell, "Towards robust automatic traffic scene analysis in real-time," in *Proc. of the 12th IAPR International Conference on Computer Vision Image Processing*, vol. 1, Oct 1994, pp. 126–131.

[6] B. Lo and S. Velastin, "Automatic congestion detection system for underground platforms," in *Proc. of International Symposium on Intelligent Multimedia, Video and Speech Processing*, 2001, pp. 158–161.

[7] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting moving objects, ghosts, and shadows in video streams," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 10, pp. 1337–1342, Oct 2003.

[8] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," in *IEEE Computer Society Conf. on CVPR*, 1999, pp. 246–252.

[9] A. M. Elgammal, D. Harwood, and L. S. Davis, "Non-parametric model for background subtraction," in *Proc. of the 6th European Conf. on Computer Vision-PartII*, ser. ECCV '00, 2000, pp. 751–767.

[10] M. Piccardi, "Background subtraction techniques: a review," in *IEEE Int. Conf. on Systems, Man, and Cybernetics*, vol. 4, Oct 2004.

[11] M. Seki, T. Wada, H. Fujiwara, and K. Sumi, “Background subtraction based on cooccurrence of image variations,” in *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, vol. 2, Jun 2003, pp. 65–72.

[12] P. Power and A. Schoonees, “Understanding background mixture models for foreground segmentation,” in *Proc. of the Image and Vision Computing*, Nov 2002, pp. 267–271.

[13] N. Oliver, B. Rosario, and A. Pentland, “A Bayesian computer vision system for modeling human interactions,” *IEEE Transactions on Pattern Analysis and Machine Intell*, vol. 22, no. 8, pp. 831–843, Aug 2000.

[14] D. Tsai and S. Lai, “Independent component analysis-based background subtraction for indoor surveillance,” *IEEE Transactions on Image Processing*, vol. 18, no. 1, pp. 158–167, Jan 2009.

[15] H. Lin, T. Liu, and J. Chuang, “Learning a scene background model via classification,” *IEEE Transactions on Signal Processing*, vol. 57, no. 5, pp. 1641–1654, May 2009.

[16] L. Maddalena and A. Petrosino, “A self-organizing approach to background subtraction for visual surveillance applications,” *IEEE Tran. on Image Processing*, vol. 17, no. 7, pp. 1168–1177, Jul 2008.

[17] I. Haritaoglu, D. Harwood, and L. Davis, “W4: real-time surveillance of people and their activities,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 809–830, Aug 2000.

[18] D. Gutchess, M. Trajkovics, E. Cohen-Solal, D. Lyons, and A. Jain, “A background model initialization algorithm for video surveillance,” in *Proc. of the Eighth IEEE Int. Conf. on Computer Vision*, vol. 1, Jul 2001, pp. 733–740.

[19] J. Jacques, C. Jung, and S. Musse, “Background subtraction and shadow detection in grayscale video sequences,” in *18th Brazilian Symposium on Computer Graphics and Image Processing*, Oct 2005, pp. 189–196.

[20] C. Jung, “Efficient background subtraction and shadow removal for monochromatic video sequences,” *IEEE Transactions on Multimedia*, vol. 11, no. 3, pp. 571–577, Apr 2009.

[21] O. Barnich and M. Van Droogenbroeck, “ViBe: A universal background subtraction algorithm for video sequences,” *IEEE Transactions on Image Processing*, vol. 20, no. 6, pp. 1709–1724, Jun 2011.

[22] W. Kim and C. Kim, “Background subtraction for dynamic texture scenes using fuzzy color histograms,” *IEEE Signal Processing Letters*, vol. 19, no. 3, pp. 127–130, Mar 2012.