Proceedings of the 32nd Chinese Control Conference
July 26-28, 2013, Xi'an, China

Object Detection Using a Moving Camera under Sudden Illumination Change

SHAKERI Moein [1], ZHANG Hong [2]

1. Department of Computing Science, University of Alberta, Edmonton, Canada
E-mail: shakeri@ualberta.ca, hzhang@ualberta.ca

Abstract: In the recent years, various background subtraction methods have been proposed and used in vision systems for moving object detection and tracking; however most of them are sensitive to illumination change and have difficulty in handling shading and shadows caused by illumination change. Although there are some algorithms to handle illumination change, they need time on the order of several frames to estimate and train the background model and, in the majority of surveillance applications, there is no such time especially when the continuous detection of moving objects after a sudden illumination change is required or if objects of interest move fast. This paper presents a robust background subtraction method which is able to cope with sudden illumination change. Our algorithm is based on the key observation that statistical background model used for object detection right after a sudden illumination change can be inferred from the model before the change sufficiently accurately to allow continued detection without delay for model re-training. The algorithm was tested on both indoor and outdoor video sequences from different datasets. Experimental results show this approach works better than the state-of-the-art algorithms in background subtraction.

Key Words: Background Subtraction, Moving Object, Sudden Illumination Change

1 Introduction

The capability of detecting moving objects from a video sequence is fundamental in many automated visual surveillance applications [1] such as traffic monitoring [2], human detection and people counting [3], and wild-life monitoring [4]. The goal of such systems is to identify the moving foreground correctly under practical working conditions. Currently, there are three main approaches to moving object detection: optical flow, temporal differencing and background subtraction, the last of which is the most common [5]. Background subtraction detects moving objects by comparing each new frame to a statistical model of the background appearance [1,5]. Pixels in an image with low probability with respect to the background model are classified as foreground. Foreground pixels can then be grouped to represent the moving objects in the scene [5].

There are many background subtraction techniques in the literature including simple techniques (e.g., based on frame differences [6], running average [7], and median filtering [8]). Some algorithms target sequences captured in changing environments due to camera noise, illumination and changing viewpoints.

In terms of the background statistical models, we can divide techniques into uni-modal (e.g., Gaussian [9] or a Chi-Square distribution [10]) and multimodal (e.g., mixtures of Gaussians [11,12], the mean-shift [13], kernel density estimation[14,15] and hidden Markov models[16]).

The existing background subtraction methods have achieved considerable success in various application domains solving many practical problems; however, many of those methods are susceptible to sudden illumination changes, which can lead to the failure of the subsequent processes, e.g. tracking and recognition [1]. Therefore it is desirable for a background subtraction technique to be able to adapt to gradual or fast illumination changes (changing time of day, clouds blocking the sun, light switching off and on, etc.), fast moving background objects (e.g. tree leaves or branches), and changes in background objects (e.g. parked cars).

The most used model is certainly the pixel-wise GMM due to its good performance in terms of robustness and the computation time, memory requirements [17]. There have been many extensions of the original GMM [11,12,18]. Most of the developed strategies use rigorous statistical models or introduce spatial and/or temporal constraints.

Recently, Barnich et al. [19] proposed a novel method for background subtraction called ViBe which is a sample-based technique that builds its model by aggregating previously observed values for each pixel location. A sample-based technique circumvents the parameter estimation step that uses past images by building a model from only observed pixel values in the current image and as a result it provides fast responses to changing events in the background by directly including newly observed values in the models. Although ViBe is fast, simple to implement and shows great accuracy on the contours of the silhouettes, it cannot cope with sudden illumination changes, as we will show later in the experiments.

In this paper, we present a novel algorithm to detect foreground objects from a real-time video that is robust with respect to change in illumination. Our algorithm is based on the key observation that the statistical model of the background after sudden illumination change can be inferred from that before the change sufficiently accurately to allow continuous detection. Specifically, in the GMM framework, the variance of a background pixel remains largely unchanged before and after illumination change whereas the mean of the mean of the background model for the pixel can be initialized to the current observed intensity value. To trigger model update, we detect the change of illumination by using a simple global image descriptor, namely, the intensity histogram. The reminder of this paper is organized as follows. Section 2 describes our proposed approach for background subtraction, Section 3 presents experiments that show the performance of the proposed algorithm comparing to some methods and finally we conclude in Section 4.

2 Proposed Method

The proposed algorithm is aimed to extract foreground objects from video sequences containing complex background; it has three main parts including building the initial background model using the standard GMM approach, detecting sudden illumination change, and reconstructing the new background model, explained in 2.1, 2.2 and 2.3 sub-sections, respectively.

2.1 Gaussian Mixture Model for Background Modeling

For completeness of discussion, this section reviews the basics of Gaussian mixture model widely used in background subtraction [18]. This technique is used to initialize the background model of a scene. This background model is able to cope with the multimodal nature of many practical situations and leads to good results when repetitive background motions, such as tree leaves or branches are encountered. Since its introduction, the model has gained vast popularity in the computer vision community [18].

We assume colored images and denote the value of a pixel at time t by $\vec{x}^{(t)}$ in a common color space (e.g., RGB or HSV). The pixel based background subtraction involves deciding if a pixel belongs to the background (BG) or the foreground (FG). The pixel is classified as background if the likelihood ratio:

$$\frac{p(BG|\vec{x}^{(t)})}{p(FG|\vec{x}^{(t)})} = \frac{p(\vec{x}^{(t)}|BG)p(BG)}{p(\vec{x}^{(t)}|FG)p(FG)} \quad (1)$$

is larger than 1 and vice versa, where $p(BG|\vec{x}^{(t)})$ and $p(FG|\vec{x}^{(t)})$ are the posterior probabilities of $\vec{x}^{(t)}$ being a part of background and foreground, respectively. In general, we do not have a prior model of the foreground nor when and how often foreground objects will be present. Therefore we assume a uniform distribution for the appearance of the foreground objects $p(\vec{x}^{(t)}|FG)$. Under this assumption, the background can be modeled by (2) [18].

$$p(\vec{x}^{(t)}|BG) > th_c = p(\vec{x}^{(t)}|FG)p(FG)/p(BG) \quad (2)$$

where th_c is a threshold value. Different methods exist for modeling the color distribution of a pixel, and a common method is GMM with M components:

$$\hat{p}(\vec{x}|X_T, BG+FG) = \sum_{m=1}^{M} \hat{\pi}_m \eta\left(\vec{x}; \hat{\vec{\mu}}_m, \hat{\sigma}_m^2 I\right) \quad (3)$$

where $\hat{\vec{\mu}}_m$ and $\hat{\sigma}_m^2$, m={1,…,M} are the estimates of the means and variances, respectively, of the component Gaussians. The identity matrix I has proper dimensions, and η is a Gaussian probability density function. Assuming that X_T is a training set with time period T and that, at time t, we have $X_T = \{x^{(t)}, \dots, x^{(t-T)}\}$.

One can estimate the background model by (4), assuming that the foreground objects are represented by components in (3) with small weights $\hat{\pi}_m$.

$$\hat{p}(\vec{x}|X_T, BG) \sim \sum_{m=1}^{M} \hat{\pi}_m \eta\left(\vec{x}; \hat{\vec{\mu}}_m, \hat{\sigma}_m^2 I\right) \quad (4)$$

If we order the M components based on the fitness value $\hat{\pi}_m/\hat{\sigma}_m$, then the first B components are used as the background model of the scene. We estimate B by

$$B = arg\min_b\left(\sum_{m=1}^{b} \hat{\pi}_m > Th_B\right) \quad (5)$$

Finally we use (6),(7) and (8) to update (3) through time.

$$\hat{\pi}_m = \hat{\pi}_m + \alpha\left(o_m^{(t)} - \hat{\pi}_m\right) - \alpha c_T \quad (6)$$

$$\hat{\vec{\mu}}_m = \hat{\vec{\mu}}_m + o_m^{(t)}(\alpha/\hat{\pi}_m)\vec{\delta}_m \quad (7)$$

$$\hat{\sigma}_m^2 = \hat{\sigma}_m^2 + o_m^{(t)}(\alpha/\hat{\pi}_m)(\vec{\delta}_m^T\vec{\delta}_m - \hat{\sigma}_m^2) \quad (8)$$

where $\vec{\delta}_m = \vec{x}^{(t)} - \hat{\vec{\mu}}_m$. Here, we fix the influence of the new samples by fixing $\alpha = 1/T$. For a new sample the ownership $o_m^{(t)}$ is set to 1 for the closest component and set to zero for all other components. The closeness of a sample to a Gaussian component is defined by the normalized distance to that component. Specifically, the squared distance to the m^{th} component is evaluated as follows [18].

$$D_m^2(\vec{x}^{(t)}) = \vec{\delta}_m^T\vec{\delta}_m/\hat{\sigma}_m^2 \quad (9)$$

We initialize our system by using the GMM method described above to build the background model. When GMM parameters converge, $\hat{\vec{\mu}}_m, \hat{\sigma}_m^2$ values are stored as the initial background model. It should be taken into account that during initialization, the scene remains static.

2.2 Detecting Sudden Illumination Changes

In order to update the background model in case of illumination change, we first need to detect the onset of an illumination change. We perform this by making use of the image intensity histogram. Specifically, sudden illumination changes can be detected using the following equation:

$$\|H_C - H_P\| > Th \quad (10)$$

where H_C and H_P are the histograms of intensity values of

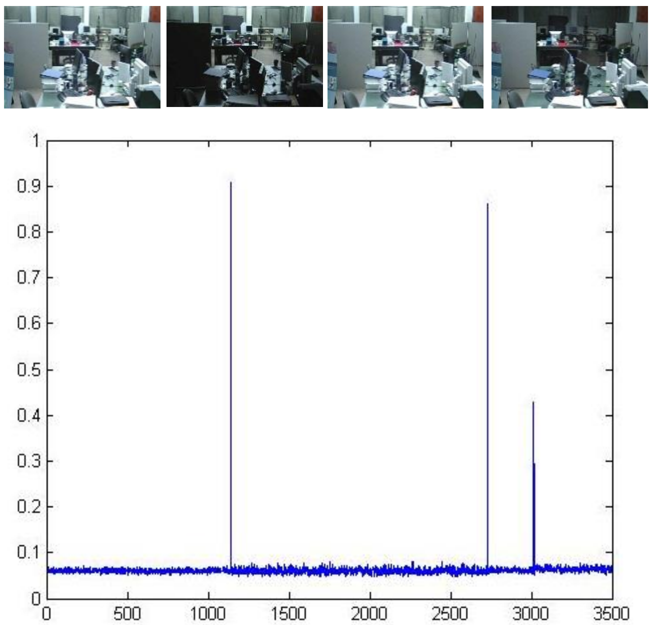


Fig. 1: Sudden Illumination Changes and relative histograms

the current and previous frame respectively and, if their difference is larger than a threshold Th, we consider that a sudden illumination change has occurred.

As an example of the reliability of this simple technique, the four images at the top of Fig. 1 represent the four lighting conditions of a video sequence. The bottom of Fig. 1 also plots the difference of histograms of two consecutive frames of the video sequence. The three frames where illumination change occurs are easily identified.

2.3 Update Background Model

There are different methods available to detect moving objects when illumination changes [20]; however most of them have training phases after the change. In addition, they are usually inefficient in terms of computation time.

In this sub-section, we present our method that re-constructs the background model promptly and efficiently upon detection of illumination change. As mentioned, our method is based on the key observation that the variance of a pixel's background model is to a large extent invariant with respect to illumination. To illustrate this observation, shown in Fig. 2 are the results of some experiments on both available different datasets and live camera to study variance changes along images obtained from Eq. 3 to Eq. 9. Clearly, the majority of the pixels retain their background variance after illumination change.

To exploit this observation, since every pixel has a corresponding variance σ^2 obtained from background training phase (described in Sub-Section A), so we will have a matrix of σ's labeled as *Sigma* in the following discussion with the same size as the size of the image. The first curve of Fig. 2 shows the similarity of *Sigmas* of Figures 3(a) and 3(b). Similarity in this case is defined as if their difference is less than a specific threshold. For instance, this similarity can be as high as to 96% if threshold is 4 and the similarity is approximately 80% at a threshold of 2.

A close look at Fig. 3 reveals that the cases of illumination change in that sequence is rather complex since Fig. 3(a) and 3(b) are actually one of the hardest cases while illumination changes. Different pixels in different parts of the scene experience different changes in their illumination level, while for some pixels illumination increases, other pixels experience reduction in illumination. Even for such a difficult situation, our key observation that variance of the background remains similar holds true. For completeness of the environments we consider in our study, Second curve of Fig. 2, shows the average similarity of *Sigma* matrixes on different datasets regarding different thresholds.

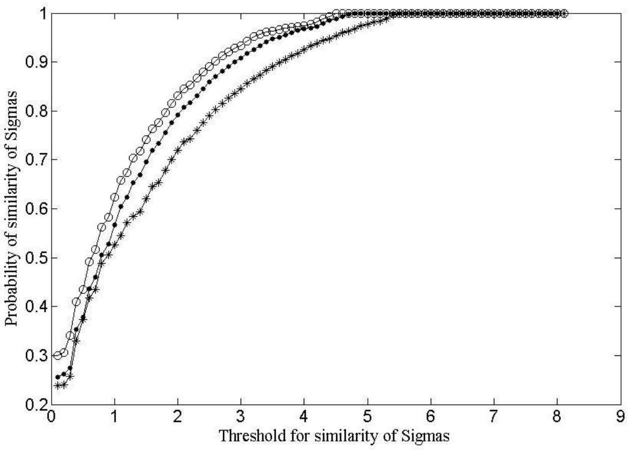


Figure 2. Similarity of Sigma on different datasets

(a) (b)

Fig. 3: (a) and (b) sample frames for sudden illumination changes

As the final example, Figure 4(a) and 4(b) depict two images taken in extremely dark and bright situations. The similarity of their *Sigma* before and after illumination change can be seen on the third curve of Fig. 2. Surprisingly approximately 75% of pixels have the same *Sigma* by considering threshold as 2. However there is always a portion of *Sigma* that are not similar, and the corresponding pixels need to be handled carefully in foreground detection, as will be described later.

Based on the above discussion, when illumination change is detected, we reconstruct the new background model based on the following equations:

$$\hat{\sigma}^2_{new} = \hat{\sigma}^2_m \tag{11}$$

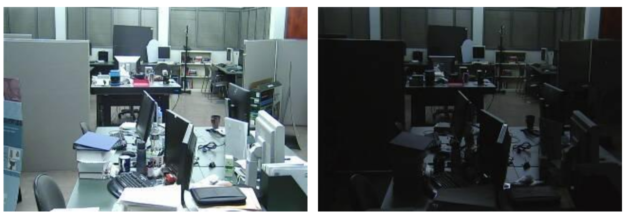
$$\hat{\vec{\mu}}_{new} = \vec{V}_{i,j} \tag{12}$$

where $\vec{V}_{i,j}$ is RGB values of the pixel (i,j) in the current frame in which the illumination has changed significantly.

Now we can estimate new parameter values in this frame by substituting $\hat{\vec{\mu}}_{new}$ and $\hat{\sigma}^2_{new}$ by $\vec{V}_{i,j}$ and previously stored $\hat{\sigma}^2_m$, and continue to detect objects without delay for training. Subsequently, for improved performance, we use Eqs. (5)-(8) to update the background model with the arrival of new images, as in the traditional background subtraction algorithm.

Using Eqs. (11) and (12), we can find out although some σs are not similar enough (remaining non-similar portion of Sigma matrixes), they converge very rapidly in comparison with the case we just use a constant value for initial value of σs.

Once the background model is replaced with the new model, moving object in the scene should be detected. Generally speaking, when the illumination changes, moving object can be either present in the scene or not. While the



(a) (b)

Fig. 4: (a) and (b) sample frames for extreme sudden illumination changes

former should be handled in specific way, in the latter there would be no special problem and the model works properly.

A difficult situation arises when the object is present in the scene while illumination changes. So that the intensity value of the object would be considered as the background according to (12). In order to overcome this difficulty and detect the object as foreground, we should reconstruct the background occluded by this object as follows.

Assuming known foreground region before any sudden change in illumination, we define a binary mask with the same size as image in which foreground (moving objects) and background pixel values are 1 and 0 respectively. We first dilate the foreground region to account for displacement of the moving objects in consecutive frames. By the use of this dilated binary mask, we can extract background behind the moving object using the original background model (see Fig.5(c)). Due to huge difference in illumination level before and after the change, we cannot substitute this extracted background as the new background model. Instead, we should follow steps below to find an appropriate background model. In these steps, the pixels behind the moving object and the other pixels in the rest of image are called masked and un-masked pixels, respectively.

- In the grayscale version of the original background (which has been stored previously), we examine all un-masked pixels in a specific neighborhood.
- For each masked pixel, for its background model, we use that of an unmasked pixel that shares the most similar intensity value with masked pixel in that neighborhood.
- Now we construct the background of the occluded region using following equations:

$$\propto = mean(\frac{I}{V_R}, \frac{I}{V_G}, \frac{I}{V_B}) \tag{13}$$

$$I\_C_{R\_new} = \propto * V\_C_R \tag{14}$$

$$I\_C_{G\_new} = \propto * V\_C_G \tag{15}$$

$$I\_C_{B\_new} = \propto * V\_C_B \tag{16}$$

where, I is intensity value of most similar unmasked grayscale pixel in original background. V is the intensity value of masked RGB pixel in original background. V_C and I_C are the intensity value of unmasked RGB pixel and estimated intensity value of masked RGB pixel in new background model.

Now we can use this I_Cs values as our new background model behind the object (see Fig. 5(d)). Fig. 5(e) shows the new background model after using I_Cs values.

3 Experimental Results

To test the proposed algorithm, we recorded several challenging image sequences with sudden illumination changes. The sequences were recorded at a resolution of 320x240, with a standard camera, in both indoor and outdoor environments. Besides, to compare our method with others, we used the dataset in [20] available on their website (http://vigir.missouri.edu/BackgroundSub.htm). The rapidly varying illumination conditions are created by randomly turning on/off different light sources.

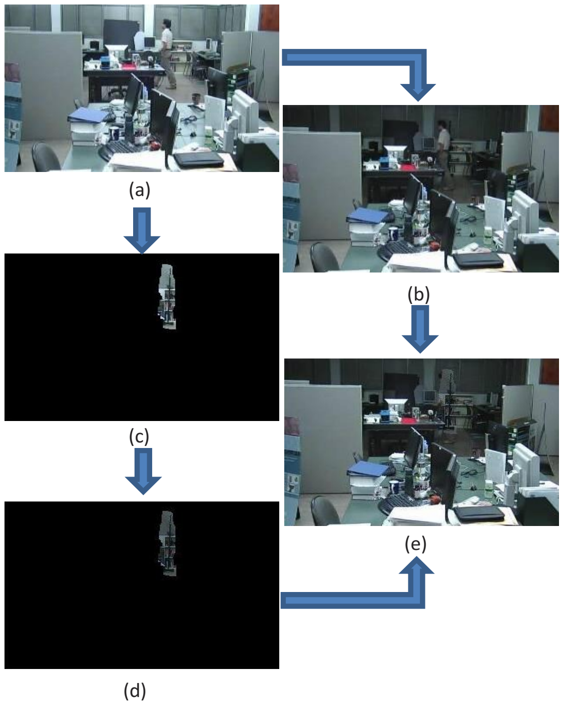


Fig. 5: Reconstructing the background model for an occluding moving object

3.1 Initialization of Parameters

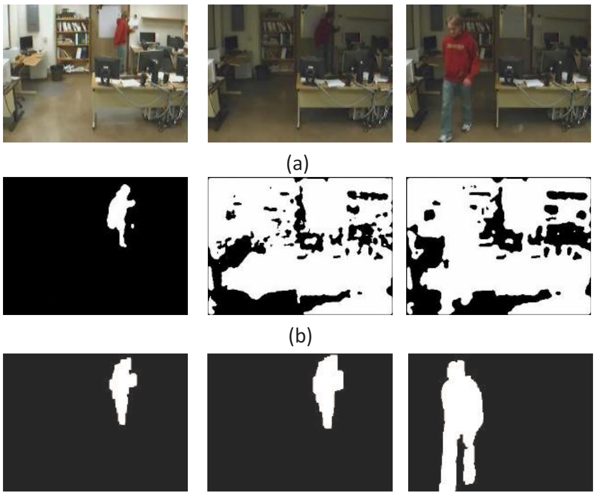
For the background model described in Section II, Step A, we assume M=3, so that the identity matrix I is 3x3, $Th_B = 0.9$, and $\alpha = 0.08$. The threshold for distance of a close component is 8. Furthermore, the dimension of means, variances and weights for each pixel is 3 by 3, 3 by 1, and 3 by 1, respectively. All images are color in RGB and evaluation of the results is performed visually by comparing the foreground objects that are detected by each of the competing algorithms.

3.2 Representative Results

Fig. 6(a) represents selected frames of the video sequence, studied in [20] which experiences high illumination change. Fig. 6(b) and 6(c) illustrate the results of the ALPCA algorithm [20] and our proposed method respectively, at a moment soon after a sudden illumination change.

Whenever any new and different (not seen before) illumination change occurs in the video sequence, the ALPCA algorithm needs to be trained and learn this new background. So if the model has not experienced any kind of illumination change beforehand, this algorithm does not work properly while training takes place as can be seen in Fig. 6(b). Although their method works well if the new illumination condition was expected so that background model could be trained beforehand; this is however an unrealistic assumption. It is important to detect a moving object correctly even if the illumination situation has not been seen in advance. Even if illumination condition can be anticipated, it is impractical to train for all various illumination levels in advance especially in outdoor environments.

Fig. 7 shows a comprehensive comparison between different approaches including ours on video sequences



(a)

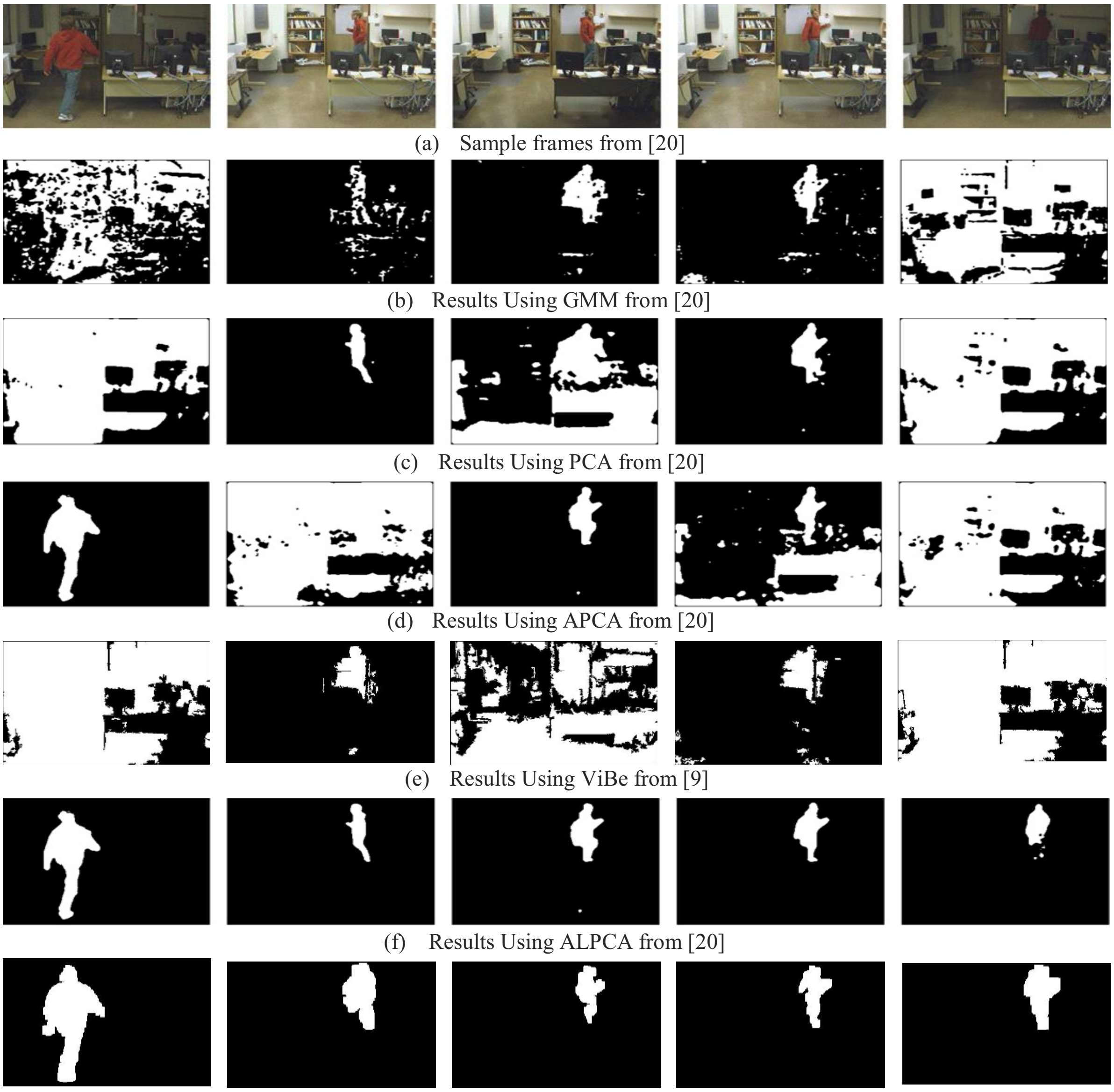
(b)

(c)

Fig. 6: (a) Samples of video frames, (b) Result from ALPCA and (c) Result of our approach

taken from (http://vigir.missouri.edu/BackgroundSub.htm). Figures 7(a), 7(b), 7(c), 7(d) and 7(f) are essentially based on reported results in [20]. Fig. 7(e) is the result of the ViBe algorithm [19] using implementation available in their website (http://www.motiondetection.org) and Fig. 7(g) is the result of our proposed method. It should be noted that although Fig. 7(f) produces almost as good a result as ours, it was obtained with training under the new illumination and considerable improvement over what it was right after lighting change shown in Fig. 7(b).

The ALPCA algorithm has been trained with this illumination situation in advance and Fig. 7(f) illustrates the results after training. However, the results of our method seem promising and comparable with theirs even without training. Furthermore, we tested our approach on a moving camera. Since in the stationary case, camera is capturing a specific scene, just one *Sigma* matrix is trained; however while working with a moving camera, it is necessary to build a separate *Sigma* matrix for each camera position (new scene). Hence there is no need for Step C of the proposed method, because its role is to find out when a sudden illumination change has been occurred. Instead, whenever camera moves to a new position, the new background model is updated using Eqs. (11) and (12).



(a) Sample frames from [20]

(b) Results Using GMM from [20]

(c) Results Using PCA from [20]

(d) Results Using APCA from [20]

(e) Results Using ViBe from [9]

(f) Results Using ALPCA from [20]

(g) Results of our Approach

Fig.7: Comparing different moving object detection method with sudden illumination changes

Fig. 8 demonstrates results on an outdoor video collected on our campus. The camera moves between two positions shown in Fig. 8(a) and Fig. 8(b). As it can be seen moving object (the walking person) is segmented correctly although when the camera returns to the first position in Fig. 8(c), illumination has changed and a big shadow of the tree has appeared in the scene. The same situation happened in Fig.8(d) while returning to the second position. The results of our method show the promise of our approach in handling significant illumination change for a moving camera returning to the previously visited position, a condition that none of the competing methods have been able to handle during the time period when they estimate the new background model.

Fig. 8: Results of proposed method in outdoor environment

4 Conclusion

In this paper we proposed an approach for background subtraction under rapidly changing illumination conditions. Our approach can robustly and effectively extract foreground objects under various illumination conditions; despite how sudden and drastic those changes may be, the proposed method can continue to detect moving objects robustly in both indoor and outdoor environments, without any training, unlike existing methods, which fail to detect objects correctly while training takes place. Extensive experiments illustrate that the proposed algorithm outperforms the state-of-the-art methods on challenging test sequences.

Acknowledgement

We are grateful to Zoran Zivkovic for publishing his code and to Olivier Barnich for ViBe implementation on the internet.

References

[1] T. Hoprasert, D. Harwood, and L. S. Davis, "A statistical approach for real-time robust background subtraction and shadow detection," in Proceedings of the 7th IEEE International Conference on Computer Vision, Frame Rate Workshop (ICCV '99), pp. 1-19, Kerkyra, Greece, September 1999.

[2] N. Friedman and S. Russell. “Image segmentation in video sequences: A probabilistic approach”, In Proc. 13th Conf., Uncertainty in Artificial Intelligence. Morgan Kaufmann, 1997.

[3] Ya-Li Hou, “People counting and human detection in a challenging situation”, IEEE Transaction on systems,man and cybernetics,Vol. 41, pp 24-33, Jan 2011.

[4] SP Yong, J. D. Deng, M. K. Purvis, Modeling semantic context for novelty detection in wildlife scenes, ICME'10, pp.1254-1259, 2010.

[5] M. Wan, X. Qin and L. He, “Background Modeling Using Mixture of Gaussians and Laplacian Pyramid Decomposition”, International Conference of Soft Computing and Pattern Recognition, Dalian, China, 2011.

[6] Ewerth, R., Freisleben, B.: Frame difference normalization: an approach to reduce error rates of cut detection algorithms for MPEG videos. In: ICIP, pp. 1009–1012 (2003)

[7] Tang, Z., Miao, Z., Wan, Y.: Background Subtraction Using Running Gaussian Average and Frame Difference. In: Ma, L., Rauterberg, M., Nakatsu, R. (eds.) ICEC 2007. LNCS, vol. 4740, pp. 411–414. Springer, Heidelberg (2007)

[8] Cheung, S.-C., Kamath, C.: Robust techniques for background subtraction in urban traffic video. In: Panchanathan, S., Vasudev, B. (eds.) Proc. Elect Imaging: Visual Comm. Image Proce. (Part One) SPIE, vol. 5308, pp. 881–892 (2004)

[9] Wren, A., Darrell, P.: Pfinder: Real-time tracking of the human body. PAMI (1997).

[10] Cavallaro, A., Steiger, O., Ebrahimi, T.: Semantic video analysis for adaptive content delivery and automatic description. IEEE Transactions on Circuits and Systems for Video Technology 15(10), 1200–1209 (2005).

[11] Stauffer, G.: Adaptive background mixture models for real-time tracking. In: CVPR (1999)

[12] Carminati, L., Benois-Pineau, J.: Gaussian mixture classification for moving object detection in video surveillance environment. In: ICIP, pp. 113–116 (2005).

[13] Comaniciu, D.: Mean shift: a robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and machine Intelligence 24(5), 603 (2002)

[14] Elgammal, A.M., Harwood, D., Davis, L.S.: Non-parametric model for background subtraction. In: Vernon, D. (ed.) ECCV 2000. LNCS, vol. 1843, pp. 751–767. Springer, Heidelberg (2000)

[15] Zivkovic, Z., Van Der Heijden, F.: Efficient adaptive density estimation per image pixel for the task of background subtraction. Pattern Recognition Letters 27(7), 773–780 (2006)

[16] Stenger, B., Ramesh, V., Paragios, N., Coetzee, F., Buhmann, J.M.: Topology Free Hidden Markov Models: Application to Background Modeling. In: Eighth Int. Conf. on Computer Vision, ICCV 2001, vol. 1, pp. 294–301 (2001)

[17] Y. Benezeth, P. Jodoin, B. Emile, H. Laurent and C. Rosenberger, “Comparative study of background subtraction algorithms”, Journal of Electronic Imaging, vol.19 (3), 2010.

[18] Z. Zivkovic, “ Improved Adaptive Gaussian Mixture Model for background subtraction”, In Proc. ICPR, 2004.

[19] O. Barnich, V. Droogenbroeck,“ViBe: A universal background subtraction algorithm for video sequences”, IEEE Transaction on Image processing, vol.20(6), pp:1709-1724, June 2011.

[20] Y. Dong, G.N. Desouza, “Adaptive learning of multi-subspace for foreground detection under illumination changes”, Computer vision and image understanding Journal, pp:31-49, 2011.