Recursive-learning-based moving object detection in video with dynamic environment

Kalpana Goyal[1] · Jyoti Singhai[1]

© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

Moving object detection is a fundamental and critical task in video surveillance systems. It is very challenging for complex scenes having slow-moving and paused objects. This paper proposes a moving object detection algorithm which combines Gaussian mixture model with foreground matching. This algorithm is able to detect slow-moving and paused objects very effectively. This algorithm uses adaptive learning rate to deal with different rates of change in background. The performance of the proposed algorithm is evaluated on the challenging videos containing strong dynamic background and slow-moving and paused objects using standard performance metrics. Experimental results show that the proposed method achieves 25% average improvement in accuracy compared over existing algorithms.

Keywords Background subtraction · Background modeling · Moving object detection · Segmentation · Foreground matching

1 Introduction

Analysis and understanding of video sequences is an active research field. The basic step in many computer vision applications like smart video surveillance, traffic monitoring, object tracking, automatic sports video analysis and gesture recognition in human-machine interface etc. is to detect the moving objects or foreground objects in the scene. Moving objects detection has also been used for wide range of applications like activity recognition, airport safety, monitoring of protection along marine border and etc. The moving object detection serves as a pre-processing step to higher-level processes, such as object classification or tracking. Hence, its performance can have huge effect on the performance of higher-level processes. The aim of moving object detection is to extract interesting moving objects in video sequences. These video sequences can have static or dynamic background. Examples of interesting moving objects are walking pedestrians and running vehicles. A frame of a video consists of two source of basic information that can be used for moving object detection and its tracking: visual characteristics or features (such as color, shape, texture) and

✉ Kalpana Goyal
   kalpana.goyal19@gmail.com

[1] MANIT Bhopal, Bhopal, India

its motion information in the consecutive frame. The moving object detection is a task to extract a low-level feature temporally and combines it to the initially segmented object to form a homogenous region in the video segment. The effort is to perceive the movement of pixels at two different instants of time and integrate them to get the relevant information. Once the intensity of pixels belonging to the certain object is sensed at different time condition, the velocity, displacement and other vectors can be computed.

Background subtraction is the most common method to identify foreground objects in the scene. In background subtraction method, each video frame is compared with a background model and the pixels whose intensity values deviate significantly from the background model are considered as foreground. Accurate foreground detection for complex visual scenes is a difficult task because real-world video sequences contain several critical situations. Some key challenges experienced in background subtraction methods are: dynamic background, moving shadows of moving objects, sudden illumination changes, camouflage, foreground aperture, noisy image, camera jitter, bootstrapping, camera automatic adjustments, paused and slow moving objects.

Previously, many researchers have proposed various moving object detection methods and different types of background models to deal with different challenges. Recursive-learning-based Gaussian Mixture Model (GMM) using foreground matching strategy is proposed to solve the problem of slow-moving and paused objects in video sequences containing dynamic background. In dynamic environment, background can be changing in some video sequences. Waving tree leaves, water rippling in a fountain and on water surface and moving curtain are main types of dynamic backgrounds.

This algorithm focus is on improving the tolerance to slow-moving and paused objects as compared to conventional GMM [19] and SAG [4]. In this algorithm, a new foreground model matching strategy is used with the standard Gaussian Mixture Model (GMM) in the proposed algorithm. In GMM and SAG , Gaussians are used to determine whether the current pixel belongs to background or foreground. However, GMM and SAG algorithm relies on the assumption that the background is visible more frequently than any foregrounds. Due to this, a long time observed moving object must be classified as the background, eventually leading to false detection results. If a slowly moving object appeared in the scene, it should not be incorporated into the background until it stopped and keep static for an enough long period.To solve the problem of slowly moving and paused objects, a simple foreground model matching strategy is used in the proposed algorithm to make a comparison. When the current pixel value fails to match the background models, foreground model is built using current pixel value as its mean and initial variance value. If the following pixel value matches the foreground model, it is classified as a foreground pixel directly. In this algorithm, focus is on improving the tolerance to low-speed objects. This method uses varying learning rate so that it can handle dynamic background.

This paper is organized as follows: section 2 gives the summary of related work, section 3 describes the proposed algorithm, section 4 discusses evaluation metrics and results obtained for the algorithm and section 5 gives the conclusion.

2 Related work

The basic technique for modeling the background uses an average [16], a median [15] or an histogram analysis over time [9]. Difference of consecutive frames [17] is also used to

detect the foreground objects. These simple techniques are very fast but their performance is poor in complex background scenes [1]. They also require a large memory and a training period where foreground objects are not present.

Clustering technique used for background modeling consists of K-means algorithm [3] and Codebook model [8]. It assumes that each pixel in the frame can be represented temporally by clusters. Incoming pixels are compared against the corresponding cluster group and the pixels matching with the cluster group are classified as background. These models seem well adapted to deal with dynamic backgrounds and noise from video compression but cannot deal with a scene which contains sudden illumination changes, slow-moving objects and moving shadows.

The kernel density estimation (KDE) [5] is a non-parametric background modeling method which estimates the density of pixels using kernel function. It is a better representation of pixels density but it requires many samples for correct estimation of the pixel density over time. Hence, it is computationally expensive and cannot be used in real-time applications [18]. This method performs well for the scenes containing static background and objects moving with constant speed. However, it cannot handle multi-modal backgrounds, slow-moving and multiple moving objects [19].

The Gaussian Mixture Model (GMM) [19] is a robust method for scenes containing background variations like multi-modal, quasi-periodic and gradual illumination changes but it cannot handle irregular background motions and sudden illumination changes [2]. Furthermore, it is a parametric model and requires tuning of parameters which makes the procedure complex, time consuming and less attractive for real-time applications. Moreover, the accuracy in selection of these parameters affects the efficiency of the system.

Several improvements have been proposed in GMM by many researchers. Kaewtrakulpong [11] used different update equations for initial training to improve the convergence rate. Lee [12] proposed a varying learning rate to improve the convergence rate and stability of GMM. Zivkovic [21] presented a GMM which uses varying number of Gaussian components. However, these algorithms cannot handle strong dynamic background, sudden illumination changes and paused and slow-moving objects.

Self-adaptive Gaussian Mixture Model (SAG) is proposed in [4] which uses varying learning rate and a global illumination change factor to deal with the sudden illumination changes but this method cannot handle strong dynamic background and paused and slow-moving objects. It uses color features which are good for high-quality video but if the background and foreground have similar color, then color features will not give good results.
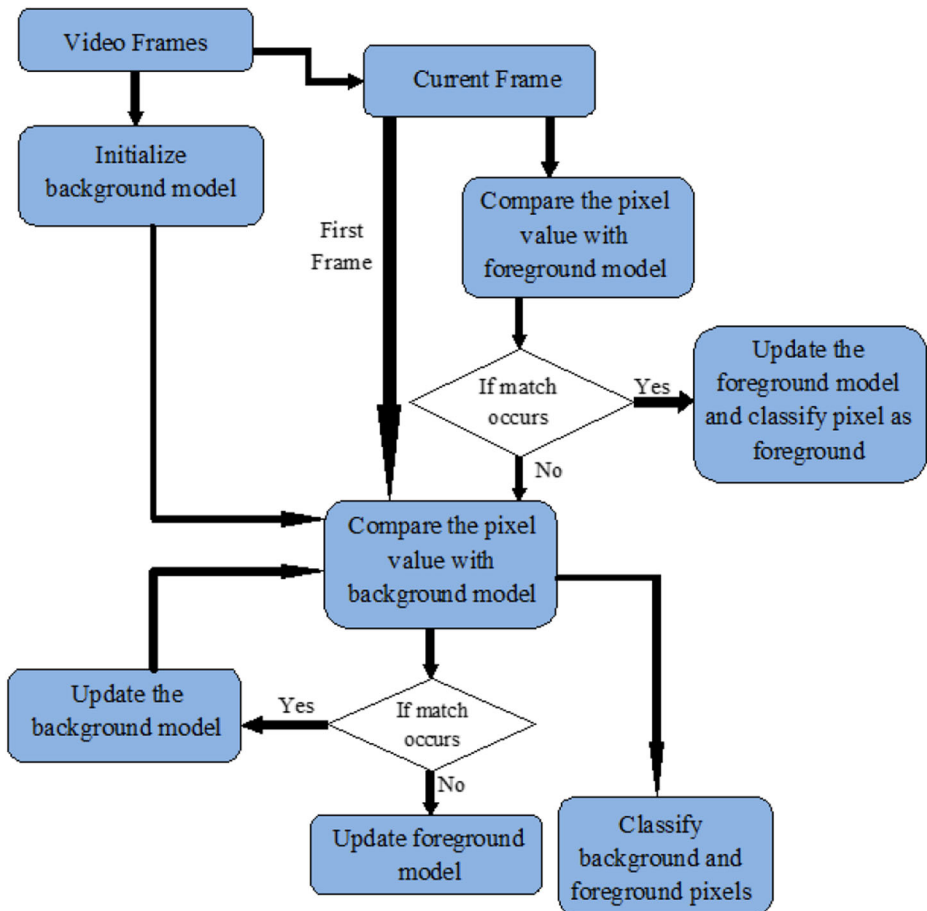
3 Recursive-learning-based Gaussian Mixture Model (GMM) using foreground matching strategy for moving object detection

Recursive-learning-based Gaussian Mixture Model (GMM) using foreground matching strategy is proposed to solve the problem of slow-moving and paused objects. This algorithm focus is on improving the tolerance to slow-moving and paused objects as compared to conventional GMM and SAG. In this algorithm, a new foreground model matching strategy is used with the standard Gaussian Mixture Model (GMM) in the proposed algorithm. In existing approaches using GMM, only background Gaussian models are used to classify the background and foreground pixels i.e., these algorithms rely on the assumption that the background is visible more frequently than any foregrounds. Due to this, slow-moving

object is classified as the background, which results in false detections. If a slowly moving object appeared in the scene, it should not be incorporated into the background until it stopped and keep static for a long period. Proposed algorithm uses both background and foreground models to make a comparison. When the current pixel value fails to match the background models, foreground model is built using current pixel value as its mean and initial variance value. If the following pixel value matches the foreground model, it is classified as a foreground pixel directly. This method uses varying learning rate so that it can handle dynamic background. Flowchart of this algorithm is shown in Fig. 1.

The recent history of the intensity values (in RGB color space) of each pixel is modeled by a mixture of Gaussian distribution. The probability of observing the current pixel value is given by the formula

$$P(X_t) = \sum_{k=1}^{K} \omega_{k,t} \times \eta(X_t, \mu_{k,t}, \Sigma_{k,t}) \tag{1}$$

Fig. 1 Flowchart of recursive-learning-based Gaussian mixture model using foreground matching moving object detection

where K gives the number of Gaussian distributions, $\omega_{k,t}$ is the weight of the k_{th} Gaussian in the mixture at time t having mean $\mu_{k,t}$ and covariance matrix $\Sigma_{k,t}$ and η is a Gaussian probability density function which is given by

$$\eta(X_t, \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}} \exp^{-\frac{1}{2}(X_t-\mu)^T \Sigma^{-1}(X_t-\mu)} \tag{2}$$

where, n is the dimension of the color space.

For computational reasons, it is assumed that the red, green and blue color components are independent and have the same variances [19]. Hence, the covariance matrix is assumed to be as:

$$\Sigma_k = \sigma_k^2 I \tag{3}$$

Due to this assumption, a costly matrix inversion is avoided at the rate of some accuracy.

A pixel matches a Gaussian distribution if its mahalanobis distance (MD) is less than a specified threshold. The squared MD is calculated as given below

$$D_k^2(x_t) = \delta_{k,t}^T \Sigma_{k,t}^{-1} \delta_{k,t} \tag{4}$$

where, $\delta_{k,t} = x_t - \mu_{k,t-1}$

For matched Gaussian distribution, parameters of that Gaussian must be updated.

$$\omega_{k,t} = (1 - \alpha)\omega_{k,t-1} + \alpha(M_{k,t}) \tag{5}$$

$$\beta_{k,t} = \alpha(1 + c_k)/c_k \tag{6}$$

$$\mu_{k,t} = \mu_{k,t-1} + M_{k,t}(\beta_{k,t}/\omega_{k,t})\delta_{k,t} \tag{7}$$

$$\sigma_{k,t}^2 = \sigma_{k,t-1}^2 + M_{k,t}(\beta_{k,t}/\omega_{k,t})\left(\delta_{k,t}^T \delta_{k,t} - \sigma_{k,t-1}^2\right) \tag{8}$$

$$c_k = c_k + 1 \tag{9}$$

where, c_k is a counter which is increased when the Gaussian parameters are updated, α is a constant learning rate and $M_{k,t}$ is set to 1 for the matched component (if the MD from the component is less than a specified threshold) and for other components, it is set to 0.

When the current pixel value fails to match the existed background models, foreground model is built using current pixel value as its mean and initial variance value. If the following pixel value matches the foreground model, it is classified as a foreground pixel directly and the pixel is used to update the foreground model. In this method, foreground models have priority over background models, the following pixels will be matched to the foreground model firstly and hence reduce the risk that foreground pixels mismatch the background models. If that pixel matches with the foreground model it will be classified as foreground pixel otherwise it will be compared to background Gaussian Models. The first B Gaussian distributions which exceed certain threshold T are considered as a background distribution:

$$B = \arg\min_b \left(\sum_{k=1}^{b} \omega_k > T\right) \tag{10}$$

A pixel is classified as background if it matches with the Gaussian distribution which is identified as background otherwise it is classified as foreground.

4 Experimental evaluation

All the experiments are done on a standard PC with 1.7 GHz Core i5 processor, 4 GB memory, and Windows 8 operating system. All the algorithms are implemented in Matlab R2013a.

4.1 Evaluation metrics

Algorithm performance is based on correctly and falsely classified pixels hence, evaluation metrics Recall (REC), Precision (PREC) and F-measure (FM) is used for quantitative evaluation. The standard performance metrics Recall (REC), Precision (PREC) and F-measure (FM) proposed in [6], are based on the accuracy in detecting the pixels as foreground and background and is measured by the number of foreground pixels classified as foreground (True Positives, TP), number of background pixels classified as foreground (False Positives, FP), number of background pixels classified as background (True Negatives, TN) and number of foreground pixels classified as background (False Negatives, FN).

Recall is also known as sensitivity and is used to measure the correctly identified foreground pixels with reference to the actual number of foreground pixels.

$$REC = \frac{TP}{(TP + FN)} \tag{11}$$

Precision is the positive predictive value and it is the fraction of correctly classified foreground pixels to the total number of pixels classified as foreground.

$$PREC = \frac{TP}{(TP + FP)} \tag{12}$$

In terms of segmentation, a low recall indicates that the algorithm has over-segmented the foreground object, while a low precision indicates an under-segmented foreground object. The F-measure measures the segmentation accuracy by considering both the recall and the precision. It is a weighted average of the recall and the precision. Higher F-measure means a good background subtraction algorithm. It is defined as

$$FM = \frac{(2 \times PREC \times REC)}{(PREC + REC)} \tag{13}$$

4.2 Results and analysis

The proposed algorithm has been evaluated on two datasets provided by [6] and [13]. The proposed approach is compared with existing background subtraction algorithms including GMM [19] and SAG [4] on these dataset. The performance of all the compared algorithms is evaluated on two aspects: dynamic background and slow-moving and paused objects. Hence, the video sequences Water Surface and Curtain from the dataset given in [13] and video sequence office from the dataset given in [6] are used to evaluate the performance of the proposed algorithm. Water surface video contains dynamic background due to rippling of water and it also contains slow-moving and paused object. The Curtain sequence contains a set of strongly waving curtains due to the wind and contains objects which becomes paused for a long time. In office video sequence, a person becomes stationary for a long time. The frame resolution of all these videos is 160×128.

To evaluate the proposed algorithm, the set of parameters used for all the experiments is: threshold=0.6, variance=6 and Mahalanobis distance D=15. Parameters used for SAG [4] are threshold=0.6, variance=11 and Mahalanobis distance D=15 and for GMM [19], threshold used is 0.25 and variance=11. For all these algorithms, learning rate used is $\alpha = 0.01$.

Optimum value (which gives higher FM) of Mahalanobis distance is selected by performing experiments with different values. If the value of MD is higher, then most of the

**Table 1** Comparison of F-measure for different values of Mahalanobis distance (D)

Video Sequences	D=10	D=15	D=20
Office	0.6086	**0.8833**	0.8286
Water Surface	0.6306	**0.9095**	0.8829
Curtain	0.6772	**0.8089**	0.7581

foreground pixels will match with background Gaussian models and these foreground pixels will be classified as background, hence recall will be very less. If the value of MD is less then background pixels will be classified as foreground, hence precision will be very less. Proposed algorithm is experimented with the different values of Mahalanobis distance (D) and F-measure of the algorithm is calculated. F-measure of Office, Water Surface and Curtain videos for $D = 10$, 15 and 20 is shown in Table 1. Highest F-measure is obtained for $D = 15$. Hence, $D = 15$ is used for all the experiments.

The detection results of the proposed algorithm for 3 frames of Office, Water surface and curtain video sequences are shown in Figs. 2, 3 and 4 respectively.

The segmentation results of GMM, SAG and the proposed algorithm on these video sequences are demonstrated for a frame in Fig. 5.

Table 2 shows the quality measures for all the algorithms on the test videos for comparison. As shown in Fig. 5 and Table 2, GMM and SAG are having less Recall and Precision than the proposed algorithm. Due to paused objects, foreground is classified as background, hence GMM and SAG results in large number of False Positives and less number of True Positives. Hence, GMM and SAG has very less Recall as compared to this algorithm. Due to dynamic background, Water Surface and Curtain video are having less Precision than proposed algorithm. Hence, GMM and SAG are having less FM than the proposed algorithm. This algorithm has improved the average FM by 39% than GMM, 19% than SAG.
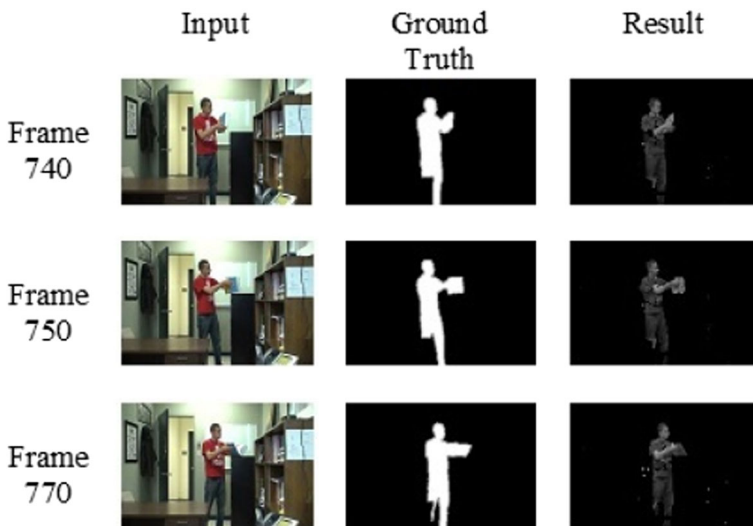
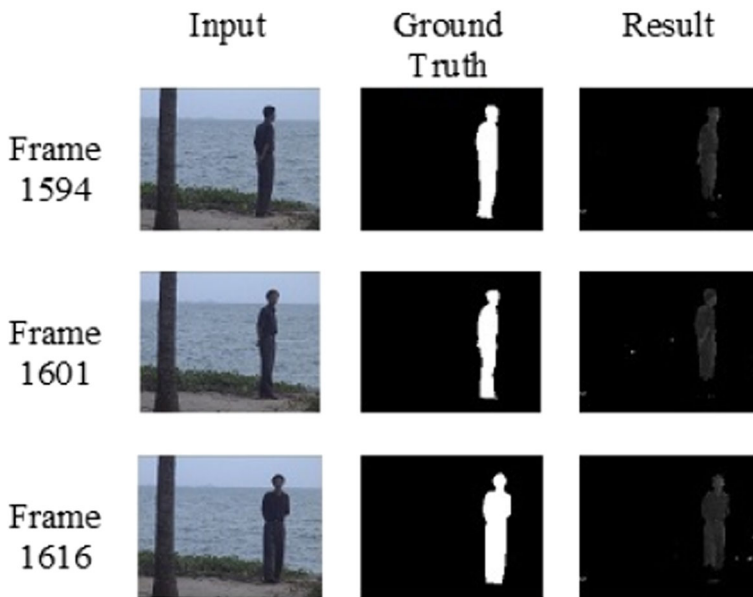**Fig. 2** Moving object detection results of 3 frames of Office video

Fig. 3 Moving object detection results of 3 frames of Water Surface video

This algorithm is also compared with KDE [5], GMM [19], SOBS [14], PBAS [7], RMoG [20] and WeSambe [10] algorithms on CDnet dataset [6]. Video sequences containing intermittent object motion are used to evaluate the performance of the proposed algorithm.
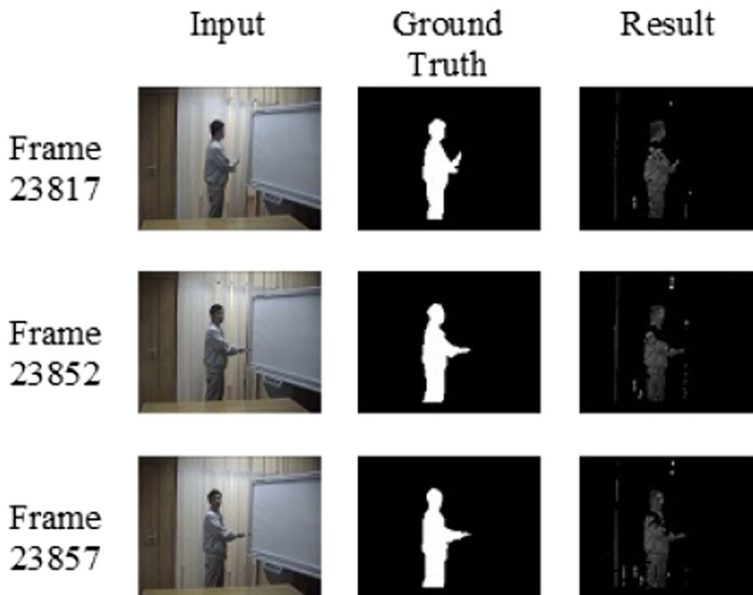

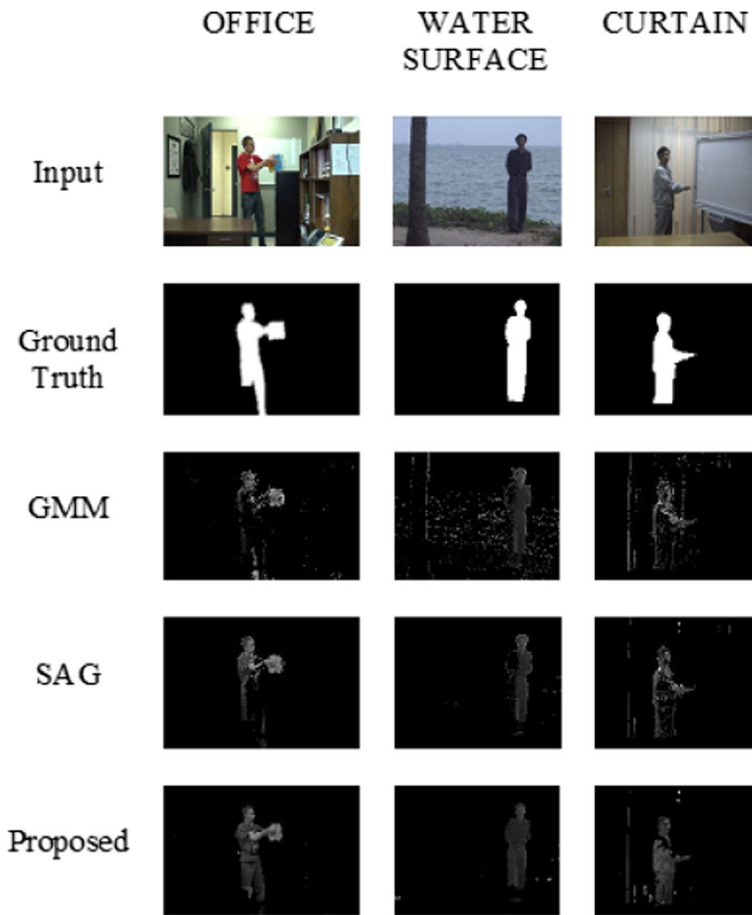**Fig. 4** Moving object detection results of 3 frames of Curtain video

Fig. 5 Moving object detection results of all the compared algorithms for the tested videos

Table 2 Comparison of quality measures for the tested videos

| Algorithms		GMM [19]	SAG [4]	Proposed
Office	REC	0.3423	0.5052	0.8268
	PREC	0.8437	0.9920	0.9480
	FM	0.4870	0.6708	0.8833
Water	REC	0.7533	0.6090	0.8637
Surface	PREC	0.6613	0.9742	0.9604
	FM	0.7043	0.7495	0.9095
Curtain	REC	0.4156	0.4802	0.8007
	PREC	0.6706	0.8669	0.8172
	FM	0.5132	0.6180	0.8089

Table 3 Comparison of the proposed algorithm with some existing algorithms using quality measures

Algorithms	REC	PREC	FM
Proposed	0.6876	0.8528	**0.7613**
WeSambe [10]	0.7472	0.7889	0.7392
RMoG [20]	0.4488	0.8026	0.5431
PBAS [7]	0.6700	0.7045	0.5745
SOBS [14]	0.7237	0.5896	0.5918
GMM [19]	0.5467	0.6458	0.5325
KDE [5]	0.5035	0.4609	0.4088

Table 3 shows the quality measures for the KDE, GMM, SOBS, PBAS, RMoG, WeSambe and the proposed algorithm. Average values of REC, PREC and FM have been calculated for the proposed algorithm for videos of intermittent object motion category while for the existing algorithms, the values of these parameters for dynamic background category are taken from www.changedetection.net [6]. Among all these algorithms, the proposed algorithm has the highest F-Measure and hence, this algorithm results in more accurate detection than existing algorithms. It has improved the F-measure by 3% with respect to WeSambe, 22% with respect to RMoG, 19% with respect to PBAS, 17% with respect to SOBS and 36% with respect to KDE.

Figure 6 Compares the total number of false classifications i.e., sum of false positives and false negatives for GMM, SAG and the proposed algorithm on the Office, Water Surface, and Curtain Sequences.

This algorithm is more efficient than existing algorithms for scenes containing dynamic background and slow-moving ad paused objects. This algorithm uses foreground matching strategy to deal with slow-moving and paused objects. Existing algorithms assume that background is visible more frequently than any foregrounds and if a slowly-moving or paused object appeared in the scene for a long time, it is classified as background which results in false detections. Hence this algorithm uses both foreground and background
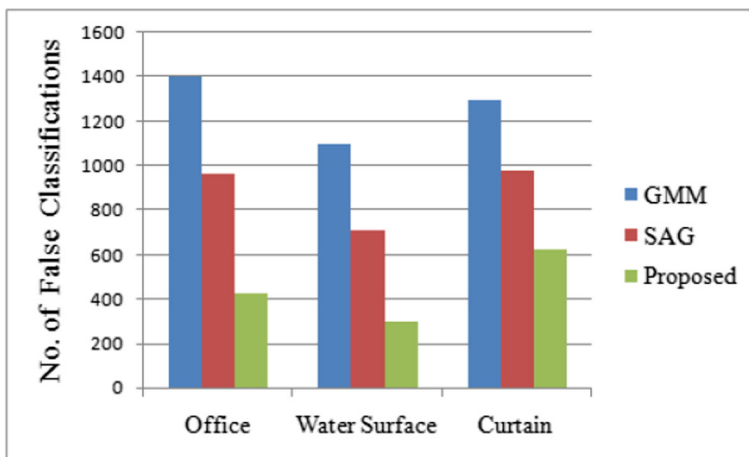
Fig. 6 Number of False Classifications

models for comparison. When the current pixel value fails to match the background models, foreground model is built using current pixel value as its mean and initial variance value. If the following pixel value matches the foreground model, it is classified as a foreground pixel directly. This algorithm uses varying learning rate to deal with the dynamic background. Learning rate changes automatically with the rate of change of background. Hence, this algorithm results in more accurate detection for scenes containing dynamic background and slow-moving and paused objects. Run time for segmentation of each frame for this algorithm is 0.066 sec. Frame rate for this algorithm is 15fps for 160×128 video frame. Computational complexity of the proposed algorithm is less than or equal to $O(NK + N)$ where, N is the number of pixels in a frame and K is the number of background Gaussian components used in the algorithm. For the existing GMM and SAG algorithm , computational complexity is exactly $O(NK)$. For the proposed algorithm it is less than $O(NK + N)$ because if any pixel value matches the foreground model, it is classified as a foreground pixel directly and it will not be matched with any of the background model.

5 Conclusion

A moving object detection algorithm is proposed in this paper which combines Gaussian mixture model with foreground matching. This algorithm detects slow-moving and paused objects very effectively. This algorithm uses adaptive learning rate to deal with different rates of change in background. The performance of the proposed algorithm is evaluated on the challenging videos containing strong dynamic background and slow-moving and paused objects using standard performance metrics. The performance of algorithm is compared with standard algorithms: GMM and SAG. The improvement in detection results by 25% on average as compared to existing algorithms concludes that the presented algorithm suitably adopts the variation in environment and is capable to effectively detect slow-moving and paused objects.

In the real time environment, there are number of challenges to face to deal with the object detection task perfectly. Some challenges are not touched in this work, which are namely shadow removal, multiple cameras video content analysis and moving camera handling. This work can be extended to alleviate these problems in future.

References

1. Bouwmans T (2011) Recent advanced statistical background modeling for foreground detection-a systematic survey. Recent Patents on Computer Science 4(3):147–176
2. Bouwmans T, Baf FE (2010) Statistical background modeling for foreground detection: A survey. Handbook of Pattern Recognition and Computer, pp 181–199
3. Butler D, Sridharan S, Bove VM (2003) J.: Real-time adaptive background segmentation. In: Multimedia and expo, 2003. ICME '03. Proceedings. 2003 international conference on, vol. 3, pp. 341–344
4. Chen Z, Ellis T (2014) A self-adaptive gaussian mixture model. Comput Vis Image Underst 122(0):35–46. http://dx.doi.org/10.1016/j.cviu.2014.01.004
5. Elgammal AM, Harwood D, Davis LS (2000) Non-parametric model for background subtraction. In: Proceedings of the 6th European Conference on Computer Vision-Part II. Springer-Verlag, London, UK, pp 751–767
6. Goyette N, Jodoin P, Porikli F, Konrad J, Ishwar P (2012) Changedetection.net: a new change detection benchmark dataset. In: Computer vision and pattern recognition workshops (CVPRW), 2012 IEEE computer society conference on, pp. 1–8

7. Hofmann M, Tiefenbacher P, Rigoll G (2012) Background segmentation with feedback: the pixel-based adaptive segmenter. In: Computer vision and pattern recognition workshops (CVPRW), 2012 IEEE computer society conference on, pp. 38–43

8. Ilyas A, Scuturici M, Miguet S (2009) Real time foreground-background segmentation using a modified codebook model. In: Advanced video and signal based surveillance, 2009. Sixth IEEE international conference on, pp. 454–459

9. Wang J, Zheng Y (2006) N.N.E.H.: Extracting roadway background image: a model based approach. Transp. Res. Rep. 1944:82–88

10. Jiang S, Lu X (2017) Wesambe: a weight-sample-based method for background subtraction. IEEE Transactions on Circuits and Systems for Video Technology, pp 1–11

11. KaewTraKulPong P, Bowden R (2002) An improved adaptive background mixture model for real-time tracking with shadow detection. In: Video-based surveillance systems, pp. 135–144. Springer US

12. Lee DS (2005) Effective gaussian mixture learning for video background subtraction. Pattern Analysis and Machine Intelligence. IEEE Transactions on 27(5):827–832. https://doi.org/10.1109/TPAMI.2005.102

13. Li L, Huang W, Gu IYH, Tian Q (2003) Foreground object detection from videos containing complex background. In: In MULTIMEDIA 03: Proceedings of the eleventh ACM international conference on Multimedia, pp. 2–10. ACM Press

14. Maddalena L, Petrosino A (2008) A self-organizing approach to background subtraction for visual surveillance applications. IEEE Trans Image Process 17(7):1168–1177

15. McFarlane N, Schofield C (1995) Segmentation and tracking of piglets in images. Mach Vis Appl 8(3):187–193. https://doi.org/10.1007/BF01215814

16. Power PW, Schoonees JA (2002) Understanding background mixture models for foreground segmentation. In: Proceedings Image and Vision Computing New Zealand

17. Roy A, Shinde S, don Kang K (2010) An approach for efficient real time moving object detection. In: Embedded systems & applications (ESA), international conference on, pp. 157–162

18. Shah M, Deng J, Woodford B (2014) Video background modeling: recent approaches, issues and our proposed techniques. Mach Vis Appl 25(5):1105–1119. https://doi.org/10.1007/s00138-013-0552-7

19. Stauffer C, Grimson W (1999) Adaptive background mixture models for real-time tracking. In: Computer vision and pattern recognition, 1999. IEEE computer society conference on., vol. 2, pp. 246–252. Los alamitos, https://doi.org/10.1109/CVPR.1999.784637

20. Varadarajan S, Miller P, Zhou H (2015) Region-based mixture of gaussians modelling for foreground detection in dynamic scenes. Pattern Recogn 48(11):3488–3503. https://doi.org/10.1016/j.patcog.2015.04.016

21. Zivkovic Z, van der Heijden F (2006) Efficient adaptive density estimation per image pixel for the task of background subtraction. Pattern Recogn Lett 27(7):773–780. https://doi.org/10.1016/j.patrec.2005.11.005