



An adaptive background modeling for foreground detection using spatio-temporal features

Subrata Kumar Mohanty¹  · Suwendu Rup¹

Received: 6 August 2019 / Revised: 7 July 2020 / Accepted: 6 August 2020 /
Published online: 7 September 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

Background modeling is a well accepted foreground detection technique for many visual surveillance applications like remote sensing, medical imaging, traffic monitoring, crime detection, machine/robot vision etc. Regardless of simplicity of foreground detection concept, no conventional algorithms till date seem to be able to concurrently address the key challenges like illumination variation, dynamic background, low contrast and noisy sequences. To mitigate this issue, this paper proposes an improved scheme for foreground detection particularly addresses all the aforementioned key challenges. The suggested scheme operates as follows: First, a spatio-temporal local binary pattern (STLBP) technique is employed to extract both spatial texture feature and temporal motion feature from a video frame. The present scheme modifies the change detection rule of traditional STLBP method to make the features robust under challenging situations. The improvisation in change description rule reflects that to extract STLBP features, the mean of the surrounding pixels is chosen instead of a fixed center pixel across a local region. Further, in many foreground detection algorithms a constant learning rate and constant threshold value is considered during background modeling which in turn fails to detect a proper foreground under multimodal background conditions. So to address this problem, an adaptive formulation in background modeling is proposed to compute the learning rate (α_b) and threshold value (T_p) to detect the foreground accurately without any false labeling of pixels under challenging environments. The performance of the proposed scheme is evaluated through extensive simulations using different challenging video sequences and compared with that of the benchmark schemes. The experimental results demonstrate that the proposed scheme outperforms significant improvements in terms of both qualitative as well as quantitative measures than that of the benchmark schemes.

Keywords Background modeling · Foreground detection · Local binary pattern · Spatio-temporal local binary pattern · Background subtraction (BGS)

✉ Subrata Kumar Mohanty
subrata@iiit-bh.ac.in

Suwendu Rup
suwendu@iiit-bh.ac.in

¹ Image and Video Processing Laboratory, Department of Computer Science and Engineering, International Institute of Information Technology, Bhubaneswar, 751003, India

1 Introduction

Video analysis and understanding is an active area of research in the field of computer vision. To detect foreground object or moving object in a scene accurately, foreground detection method serves as a basic step in many computer vision applications like visual surveillance [1, 19, 25, 30, 40, 47, 54, 63], human-machine interaction [33, 52, 56, 59], object tracking [22, 28, 34], vision-based hand gesture recognition [48], person counting [29, 38], traffic surveillance [12, 17, 23, 31, 37, 42, 50], content-based video coding [11], optical motion capture [6]. This pre-processing task further helps the higher level processes for better object classification and tracking. Hence, an accurate foreground detection with minimum false error rate is very crucial for the performance of higher-level processes. The basic method for detecting moving objects in a video sequence is the background subtraction. In background subtraction technique, it is necessary to build and maintain an up-to-date background model, where each pixel from next frame is compared with the background model. When the pixels from the next frame are deviate from that of the background model are labeled as moving object. It is a difficult task to create an accurate foreground detection in complex environment where few key challenges experienced as background subtraction methods are dynamic background, sudden/gradual changes in illumination, noisy image, camouflage and camera jitter.

Many significant efforts in this direction are made to achieve high accuracy for moving object detection under complex environments [5]. Foreground detection is broadly realized by three approaches namely background subtraction approach, frame-difference approach and optical flow approach. Among these, the background subtraction approach is most widely utilized technique in background modeling [61]. Besides the aforementioned approach, there exist several model-based approaches for robust foreground detection like HOG detector [14], DPM detector [2] etc. However, the above mentioned detectors are always well suited for specific applications and may fail in general cases. To mitigate this issue, a popular feature extractor namely STLBP came into existence, which potentially employed in several reported works for foreground detection under challenging real time applications. STLBP is a variant of local binary pattern (LBP) where each pixel is modeled as a group of STLBP dynamic texture histogram which collectively captures both spatial texture features as well as temporal motion features from a video frame [64]. However, the overall foreground detection accuracy of STLBP degrades in strong multimodal background conditions. This is due to the fact that, the texture features extracted in STLBP method are sensitive towards the dynamic behavior and illumination variation of background. Further, during background modeling a constant learning rate and a threshold value is adopted that leads to an increase in false detection rate. In the suggested scheme, a STLBP feature-based robust foreground detection algorithm is proposed for background modeling. The key contributions of the suggested scheme are outlined as follows:

- I A modification in change description rule for traditional STLBP method is proposed in foreground detection to obtain the robust features under complex situations.
- II Next, an adaptive strategy is formulated in background modeling to compute the learning rate (α_b) and threshold value (T_p).
- III The efficacy of the proposed scheme is validated in terms of qualitative and quantitative measures with that of the state-of-the-art approaches.

The remaining sections of the article is structured as follows. Section 2 elaborately presents the relevant works for foreground detection. In Section 3, some preliminaries about

the LBP and STLBP techniques are discussed. Section 4, puts forward the proposed scheme for foreground detection. Experimental results and discussions are critically analyzed in Section 5. Finally, Section 6 provides the concluding remark.

2 Related work

In recent past, several research findings on foreground detection are reported [5, 49, 53]. In 2004, Piccardi briefly categorized the methods used for foreground detection based on accuracy, speed and memory requirements [49]. The suggested scheme helps to compare the complexity of different methods and allow the readers to select an adaptive method for specific purpose. In this work an efficient background modeling strategy is adopted under challenging conditions like illumination variation, dynamic background, camouflage, rippling water and noise. Bouwmans et al. in [4] provided a detailed survey on statistical modeling used for foreground detection. In most of the earlier schemes, foreground is extracted by subtracting the reference/background frame from the current frame. The reference frame can be taken as (1) first frame or previous frame [51], (2) average of several consecutive frames [36], (3) median of several consecutive frames [39]. The above mentioned methods claim to be very simple and fast for foreground detection. However, this is very sensitive to background changes which in turn leads to degrades in overall performance.

To overcome the limitation of basic background subtraction method, Wren et al. [59] proposed a simple Gaussian distribution to model the intensity value of a pixel at a fixed location over time. Further, the authors have used an adaptive updating technique for background modeling under dynamic background. However, this uni-modal Gaussian distribution model is not suitable for multi-modal background. So to resolve this issue, a multi-modal Gaussian distribution model has been proposed to provide the solution over uni-modal Gaussian distribution.

In 1999, the most widely accepted multi-modal distribution over time is proposed by Stuffer and Grimson [55] which is popularly known as mixture of Gaussians (MoG) or Gaussians mixture model (GMM). In this work, the background model is adaptively updated in an iterative manner and performs significantly well under gradual illumination changes and non-stationary background conditions. The GMM model provides the best solution to many practical situations for foreground extraction. There are numerous improvements of GMM model has been incorporated in literature [8, 58, 65]. However, when background varies very fast (many modes, i.e. more than 5 modes), a few Gaussians (usually 3 to 5) are not sufficient to accurately model the background. GMM also needs appropriate threshold value and learning rate for obtaining improved results.

To overcome the problem of GMM, in 2000, A. Elgammal et al. [16] proposed a non-parametric approach based on kernel density estimator (KDE). This approach measures the probability distribution of background values by taking the recent history at each pixel value over time. The distribution is then used to map the pixel belongs to foreground or background. The major limitations associated in KDE is the huge computational time as it takes the history of a pixel to measure the density.

Codebook method is another notable approach to model the background [35]. In this work, each pixel comprises a codebook and each codebook consist of several codewords. The number of codewords comprise the variation in pixel. The improvement over this scheme reported in [60] uses both spatial and temporal information. This method can able to detect the moving object under dynamic environment and decreases the false positive rate to a certain extent.

In [26], M. Heikkilä et al. proposed a texture-based approach for background modeling using a group of histogram-based on local binary pattern (LBP). The most important property of LBP is the robustness behavior towards the illumination changes and its computational simplicity. This method exhibits improved result under slow moving background pixels. However, its performance degrades sharply when backgrounds have strong changes (under multi-modal background condition).

Zhang et al. [64] extended the algorithm in [26] by measuring the local binary pattern in spatio-temporal domain named as spatio-temporal local binary pattern (STLBP). For modeling each pixel, the authors use weighted LBP histograms which includes both spatial and temporal information. The limitation of this approach is that, it uses constant learning rate which reduces the detection performance when the speed of a moving object varies across the successive frames. In [3], the author proposed a local binary self-similarity pattern and demonstrate its effectiveness against state-of-the-art methods.

There exist several foreground detection algorithms based on subspace learning models such as Principal Component Analysis (PCA) [7]. In PCA, the background sequence is modeled by low-rank matrix and sparse error to constitutes the foreground objects. In particular, PCA-based BGS techniques are robust against noise and illumination changes. Several variants of PCA-based techniques in the field of change detection are designed and developed by many researchers [10, 57]. The major challenges associated in PCA-based BGS is the absence of any update mechanism for classification of background/foreground. Further, the overall execution time is quite high for long duration video sequences [7]. In 2013, Feng et al. [18] proposed an online Robust PCA (OR-PCA) to reduce the memory cost and enhance the computational efficiency. Furthermore, Javed et al. [32] improved the original OR-PCA for an accurate foreground detection using image decomposition technique. In this work, the input video frames are decomposed into Gaussian and Laplacian images. Then, OR-PCA is applied to both the decomposed image with specific parameters to model the background separately. Then, an alternate initialization process is applied to speed-up the optimization process. Again, Han et al. [24] improved the OR-PCA for background subtraction considering camera jitter condition.

In 2018, Goyal et al. [20] proposed a counter-based adaptive background learning with respect to changes in background. In this work, the authors have addressed the Heikkilä technique [26] for foreground detection by updating the learning rate by considering different challenging conditions. However, this method achieves lower accuracy with respect to different conditions like camouflage, shadow, outdoor varying illumination environments and high dynamic background. In [62], authors utilize the advantage of intensity, color and Local Ternary Pattern (LTP) features for background modeling. In this scheme, the authors have obtained five features for each pixel. Again, out of five features, the dominant feature is calculated adaptively over the pixel in time domain for the improvement of accuracy in detection of foreground under complex backgrounds. Again, the authors use double threshold value for further improvement in accuracy of detection. In 2019, Moudgollya et al. [41] used angle co occurrence matrix (ACM) along with gray level co-occurrence matrix (GLCM) for extraction of six features. In this work, the authors have employed the merit of ACM and GLCM for foreground detection under different challenging environments. The major drawback of this method is the huge computational overhead.

In 2017, Z. Zhong et al. [66] proposed a robust background modeling method, which updates the background at pixel-level and object-level to detect the motionless or slow-moving objects. This method is effectively works well in many real time application, particularly when there is a variation of speed of foreground objects. In 2018, Panda et al. [45] proposed a spatio-temporal BGS method utilizing Wronskian change detection model.

Further, they have used a new fuzzy adaptive learning rate to improve the detection accuracy. Again in [46], the authors have used multi-channel Wronskian change detection model and codebook technique for foreground detection. As these methods are effective to provide impressive results under challenging environments, but it fails to reduce the false detection rate under dynamic environments. Its due to the fact that the aforementioned schemes [45, 46] utilizes of a constant threshold value for all type of complex situations in a video sequence.

From different literature findings it has been noticed that most of the schemes for foreground detection operates in spatial domain instead of both spatial as well as temporal domain. Further, in background modeling most reported schemes use a constant learning rate and a constant threshold value. This arbitrary selection of learning rate and threshold value does not ensure improved results for all challenging situations. So the aforementioned facts motivate us to propose a modified STLBP-based technique with adaptive formulation of learning rate and threshold value for foreground detection.

3 Overview of LBP and STLBP methods For BGS

This section presents an overview of LBP and STLBP-based approach for background subtraction. To give a better insight the following subsections are presented as explained below.

3.1 Conceptual overview of local binary pattern

Local binary pattern (LBP) is a gray scale invariant in nature and a powerful texture descriptor [26, 43, 44]. LBP is a well accepted feature extractor in many pattern recognition and computer vision applications. The prime usefulness of LBP operator is that it is very robust to monotonic transformation of gray value of the pixel and gives an excellent information about the region texture feature.

LBP operator labels the neighborhood of a pixel using binary derivative technique. The binary derivative of a pixel in a given image region can be measured by thresholding the relation of neighborhood pixels with that of center pixel. The classical or traditional LBP operator works in a 3×3 local region of an image. It generates a binary value 1 when a neighboring pixel is greater than or equal to the center pixel and generates binary value 0, if neighboring pixel is less than that of the center pixel. The mathematical expression of LBP is expressed as follows:

$$LBP(x_c, y_c) = \sum_{i=0}^7 s(g_i - g(x_c, y_c)) \times 2^i, \quad i = 0, 1, 2, \dots, 7 \quad (1)$$

where, g_i represents the gray value of surrounding pixels with respect to center pixel coordinate (x_c, y_c) . The function $s(g_i - g(x_c, y_c))$ is defined as,

$$s(g_i - g(x_c, y_c)) = \begin{cases} 1 & \text{if } g_i - g(x_c, y_c) \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

The 8 neighbors of the center pixel in 3×3 region are then represented with a 8 bit number. Thereafter, the binary number is converted to its equivalent decimal value, which presents the LBP label for the center pixel. The basic working principle of LBP operator is shown in Fig. 1.

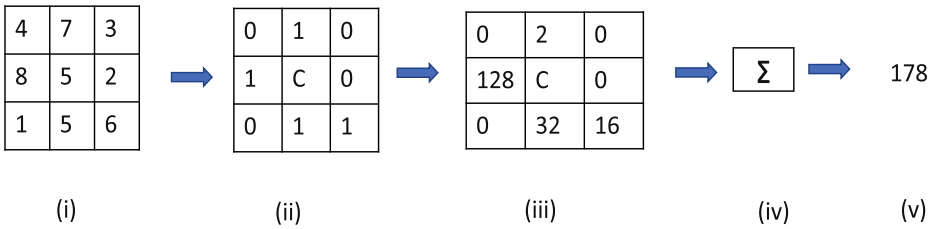


Fig. 1 Example of Basic LBP operator. (i) 3×3 region of an image, (ii) binary conversion after thresholding, (iii) Weights assigned to neighborhood pixels (equivalent decimal value of the binary number in clock wise direction starting from pixel coordinate(1,1)), (iv) Summing of weights, (v) LBP code for center pixel

Ojala et al. [44], extended the definition of basic version of LBP in which I number of equally-spaced pixels located on a circle of radius R to measure the LBP code for the center pixel as shown in Fig. 2. The value of I may be any number unlike fixed number of neighboring pixels (i.e. 8) used in basic LBP operator. If the number of pixels I is less it leads to less computational overhead. On the otherhand, large number of pixels I require in large-scale texture features. The circular LBP can be mathematically expressed as,

$$LBP_{I,R}(x_c, y_c) = \sum_{i=0}^{I-1} s(g_i - g(x_c, y_c)) \times 2^i, \quad i = 0, 1, 2 \dots I - 1 \quad (3)$$

where, $g(x_c, y_c)$ is the center pixel value with coordinates (x_c, y_c) of circular region of radius R and g_i ($i = 0, 1, 2 \dots I - 1$) is the gray values of the equally spaced I pixels on the circle of radius R centered at (x_c, y_c) . The definition of $s(g_i - g(x_c, y_c))$ is same as in (2). The (I, R) values may be varied like (4,1), (8,1), (16,2) etc proposed in [44]. The coordinates of circular neighbor pixels is given as,

$$[x_i, y_i] = \left[x_c + R \cos\left(\frac{2\pi i}{I}\right), y_c - R \sin\left(\frac{2\pi i}{I}\right) \right], \quad i = 0, 1, 2 \dots I - 1 \quad (4)$$

The coordinates of the neighbors that do not fall on pixel points are approximated through bilinear interpolation [44].

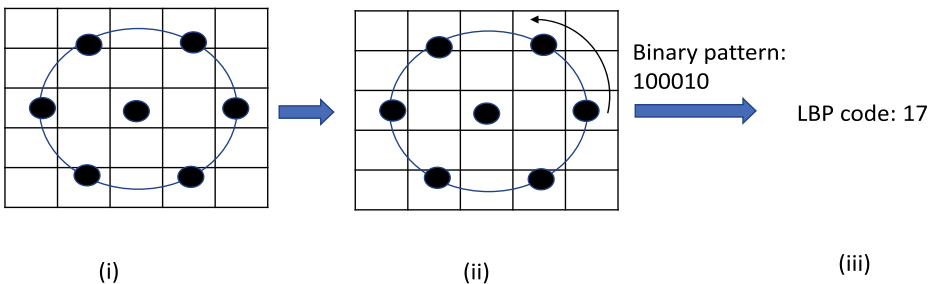


Fig. 2 LBP code for circular neighbor (6,2)

3.2 Background subtraction using LBP method

In 2006, Heikkila et al. [26] proposed an efficient approach for moving object detection by modeling the background using region texture-based method. This method, makes the circular LBP more robust against small or negligible changes in pixel value. Here, the circular LBP operator has been modified as,

$$LBP_{I,R}(x_c, y_c) = \sum_{i=0}^{I-1} s(g_i - g(x_c, y_c) + \alpha) \times 2^i \tag{5}$$

where, expression for $s(x)$ is same as defined in (2) and α is responsible to retain the discriminative power of LBP operator when the gray values of neighboring pixels are very close to the center pixel value of a region [26].

The limitation of this method is that it does not contain any temporal information of the pixels. As a result, it fails to detect moving object accurately under dynamic environmental conditions.

3.3 Background subtraction using STLBP method

To mitigate this aforementioned problem, Zhang et al. [64] extended the work reported in [26] by presenting a new dynamic texture operator namely spatio-temporal LBP (STLBP).

From the Fig. 3, let F_t and F_{t-1} be the current and previous frame respectively. In current frame F_t , $g(x_c, y_c; t)$ and $g(i; t)$ ($i = 0, 1, 2, \dots, I - 1$) are the center pixel and circular neighbor pixels respectively. To measure the motion information along the temporal direction, only circular neighbors $g(i; t - 1)$, ($i = 0, 1, 2, \dots, I - 1$) of frame F_{t-1} is considered. Two LBP codes for center pixel $g(x_c, y_c; t)$ for spatial and temporal neighboring pixels are expressed as follows:

$$LBP_{I,R}(x_c, y_c; t) = \sum_{i=0}^{I-1} s(g(i; t) - g(x_c, y_c; t) + \alpha) \times 2^i \tag{6}$$

$$LBP_{I,R}(x_c, y_c; t - 1) = \sum_{i=0}^{I-1} s(g(i; t - 1) - g(x_c, y_c; t) + \alpha) \times 2^i \tag{7}$$

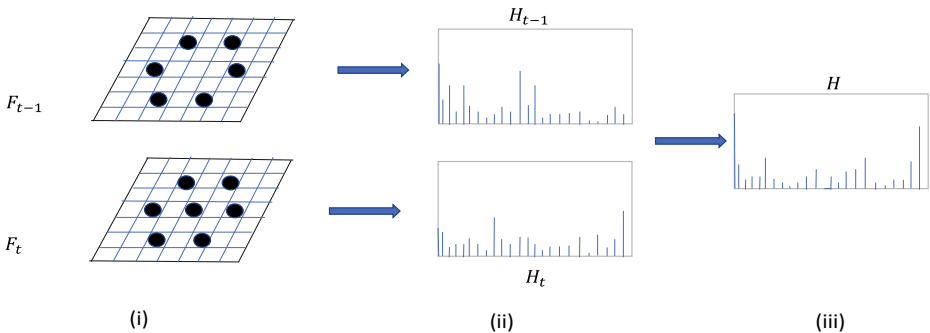


Fig. 3 STLBP histogram measurement for circular neighbor. (i) Circular neighbor of two consecutive frames with same coordinate values with respect to center pixel, (ii) Histogram measurement on 2^I bins of each frame and (iii) Resultant histogram

$LBP_{I,R}(x_c, y_c; t)$ and $LBP_{I,R}(x_c, y_c; t - 1)$ represents the spatial LBP and temporal LBP of a circular region having radius R . The spatial histogram (on current frame) and temporal histogram (on previous frame) are computed from spatial LBP using (6) and (7) respectively over a circular region of radius R_{region} . Figure 3 illustrate the STLBP histogram measurement of a circular neighbor. Next, the STLBP histogram is calculated by summing the spatial histogram and temporal histogram as follows:

$$H_t = \omega \times H_{t-1,i} + (1 - \omega) \times H_{t,i}, \quad i = 0, 1, 2, \dots, I - 1 \tag{8}$$

where $H_{t,i}$ and $H_{t-1,i}$ are the current and previous frame histograms of i^{th} bin respectively shown in Fig. 3. H_t is the resultant spatio-temporal histogram at time t and ω is the spatio-temporal rate. Each pixel is modeled with a group of adaptive STLBP-based histogram techniques which are computed over a circular region of radius R_{region} .

The advantage of STLBP method is that it extracts the spatial information as well as temporal information for each pixel that further reduces the false detection rate under dynamic condition. From different observation it is noticed that a constant threshold and learning rate is used in STLBP method. As a result the overall performance of foreground detection accuracy reduces to a great extent.

4 Proposed approach

In the proposed work, an improvement in spatio-temporal LBP (STLBP) descriptor [64] for foreground detection under complex scenario is proposed. In the suggested scheme, a change description rule is formulated which essentially modifies the traditional STLBP method reflects in (6) and (7) respectively. The modified change detection rule is defined as follows:

$$LBP_{I,R}(x_c, y_c, t) = \sum_{i=0}^{I-1} s_{T_r}(g(i; t) - \mu_g(t) + \alpha) \times 2^i \tag{9}$$

$$LBP_{I,R}(x_c, y_c, t - 1) = \sum_{i=0}^{I-1} s_{T_r}(g(i; t - 1) - \mu_g(t) + \alpha) \times 2^i \tag{10}$$

where,

$$s_{T_r}(x) = \begin{cases} 1 & \text{if } x \geq T_r \times F_t(x_c, y_c) \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

and

$$\mu_g(t) = \frac{1}{I} \sum_{i=0}^{I-1} g(i; t) \tag{12}$$

T_r ($= 0.2$) is the user defined constant value.

To support our claim for the proposed foreground detection with the modified change description rule, two experiments are carried out as follow: The first experiment shows the variation in feature under dynamic environment which is shown in Fig. 4. Further, second experiment focuses on variation in illumination (refer Fig. 5). Here, the mean of surrounding pixels is taken instead of considering the center pixel across the local circular region of radius $R = 2$ to generate spatial-LBP which is defined in (9). The reason behind the selection of mean of the surrounding pixel with that of the center pixel is that, the obtained mean values of surrounding pixels for two consecutive frames with high dynamic region

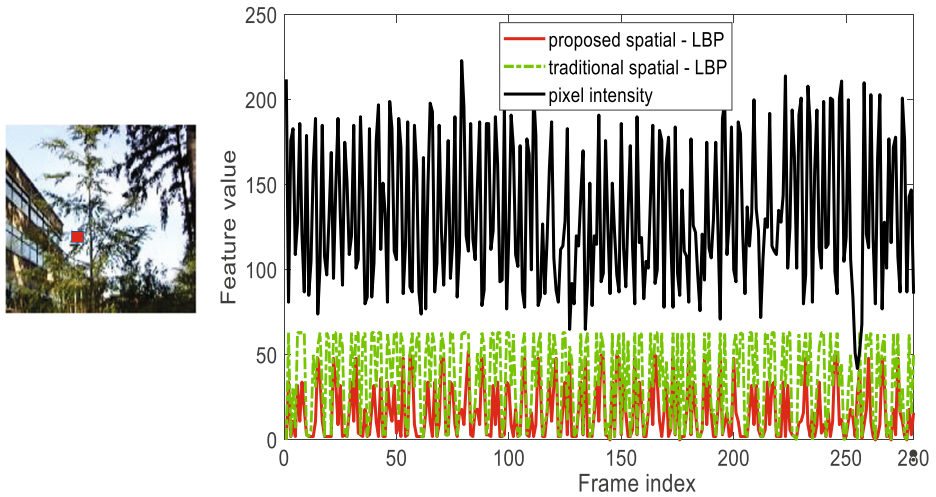


Fig. 4 Feature variation of a particular pixel (65, 77) on WT dataset across 280 frames

are invariant in nature as compared to the variation of center pixel of these regions. So that motivate authors to extract the invariant features under dynamic and illumination variation environments. During the experiment, two widely accepted video sequences namely WT (Waving Tree) [9] and OTCBVS [15] with 280 frames are used. An illustration for feature variation of a particular pixel with coordinate (65, 77) on WT (Waving Tree) and a particular pixel with coordinate (128, 85) on OTCBVS datasets across all 280 frames are shown in Figs. 4 and 5 respectively.

From the Figures it has been noticed that in both the video sequence (WT and OTCBVS) the proposed spatio-temporal LBP features value give less variation with respect to the erratic behavior of a dynamic pixel at a particular location along 280 frames. Additionally, the proposed scheme shows less variation in feature values in temporal direction than that of the traditional STLBP scheme.

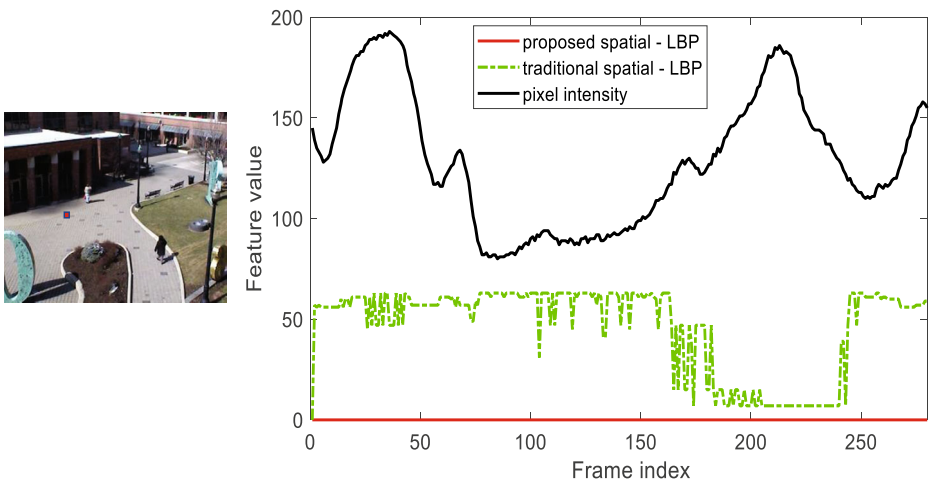


Fig. 5 Feature variation of a particular pixel (128, 85) on OTCBVS dataset across 280 frames

Additionally, to show the effectiveness of proposed foreground detection, another experiment is conducted between traditional spatial-LBP expressed in (6) and proposed spatial-LBP expressed in (9) is shown in Fig. 6. During experimentation, initially the texture features are extracted from both the traditional spatial-LBP and proposed spatial-LBP. Next, the extracted texture features from both the techniques are given as the input to GMM [55]. The parameters of GMM are kept same for both the inputs. Further, the performance measures like *Recall*, *Precision* and *F – measure* resulted from GMM process are computed. Figure 6 shows the comparison between the encodings of traditional spatial-LBP and proposed spatial-LBP for Highway [21], PETS 2006 [21] and OTCBVS [15] datasets respectively. From Fig. 6, it is clearly noticed that the proposed spatial-LBP achieves potential improvements as compared to traditional spatial-LBP with respect to the aforementioned performance measures.

Further, Fig. 7 shows the generated foreground of the proposed spatio-temporal LBP over traditional spatio-temporal LBP on different threshold value. In this comparison, we adopt the background modeling procedure reported in [26]. All parameters ($\omega = 0.01, \alpha_b =$

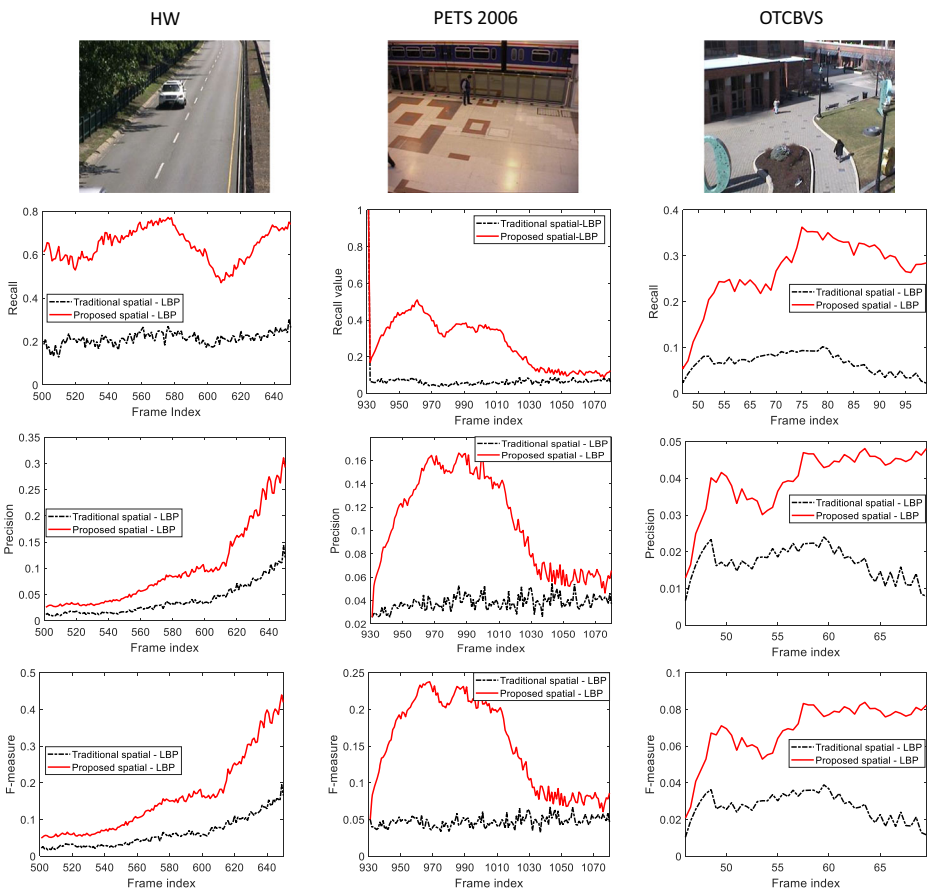


Fig. 6 An experimentation to show the comparison between the encodings of traditional spatial-LBP and proposed spatial-LBP for highway (HW), PETS 2006 and OTCBVS video sequences. The texture features from traditional spatial-LBP and proposed spatial-LBP are considered as input to GMM. The second, third and forth rows show the *Recall*, *Precision* and *F – measure* values respectively

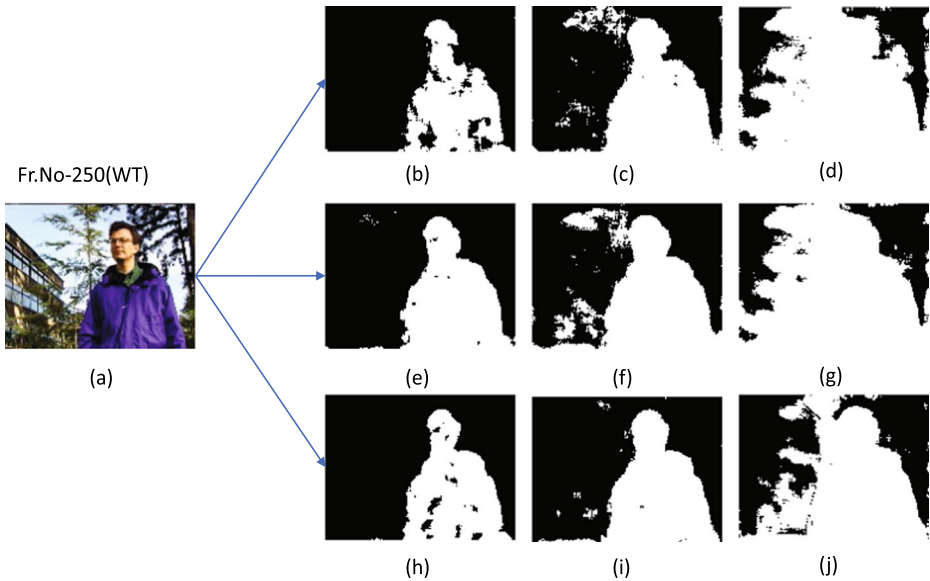


Fig. 7 Foreground segmentation results comparison on traditional STLBP and proposed STLBP based on different threshold (T_p). **a** 250th frame of WT video sequence, **b–d** segmented foreground results of traditional STLBP method [64] with T_p value 0.6, 0.7 and 0.8 respectively, **e–g** segmented foreground results of modified STLBP method with T_p value 0.6, 0.7 and 0.8 respectively, **h–j** segmented foreground results of the proposed STLBP method with T_p value 0.6, 0.7 and 0.8 respectively. Both traditional STLBP and proposed STLBP are input to the background modeling technique reported in [26]

$\alpha_w = 0.01, T_B = 0.8, \alpha = 3$ and $T_r = 0.2$) are kept fixed for fair comparison in all methods. In Fig. 7, (b)–(d) show the segmented results of foreground by traditional spatio-temporal LBP method as in [64], (e)–(g) show the segmented results for foreground using the extracted STLBP features by considering mean of the surrounding pixels instead of center pixel shown in (6) and (7). Finally, (h)–(j) show the segmented results of moving objects by proposed spatio-temporal LBP method with the changed description rule expressed in (9) and (10). The results for each methods are shown on the basis of threshold values (T_p) 0.6, 0.7 and 0.8 respectively.

Next, we compute the two histograms H_t and H_{t-1} of the binary pattern resulted from a circular region depicted in (9) and (10). This histogram of binary pattern over a region is used as local texture feature. As shown in Fig. 3, the H_t and H_{t-1} over a region is computed as follows:

$$H_{t,n} = \sum_{x,y \in R} N(LBP_{I,R}(x, y; t) = n), \quad n = 0, 1, 2, \dots, 2^I - 1 \tag{13}$$

and

$$H_{t-1,n} = \sum_{x,y \in R} N(LBP_{I,R}(x, y; t - 1) = n), \quad n = 0, 1, 2, \dots, 2^I - 1 \tag{14}$$

where $H_{t,n}$ and $H_{t-1,n}$ represents the n^{th} bin histogram of t and $t - 1$ respectively; and $N(z)$ is 1 (if z is true) or 0 (otherwise). The histogram H of STLBP is generated with the help the aforementioned two histograms as follows:

$$H_t = \omega \times H_{t-1,n} + (1 - \omega) \times H_{t,n}, \quad n = 0, 1, 2, \dots, 2^I - 1 \tag{15}$$

The STLBP histogram expressed above provides the information about spatial texture feature and temporal motion feature. The parameter ω depends on the degree of dynamic background condition. If the background changes very fast then larger value of ω is adopted otherwise a small value of ω is considered for small changes in background. In our case the value of ω is taken as 0.01 experimentally.

4.1 Proposed background modeling algorithm using STLBP feature

Background modeling is considered to be a prime step for background subtraction process as it has a higher impact on overall detection accuracy. For background model initialization, the STLBP histogram is evaluated over a circular region of radius called R_{region} centered at a particular pixel (x, y) using first and second frame. According to the MOG, the K background models for a pixel consists of $\{H_{m_k}^{(x,y)}, k = 1, 2...K\}$. Initially, the weights of each model histograms are taken as equal value. Next, the weights of the model histograms w_k ($k = 1, 2...K$) are normalized to make their sum equal to 1. From the next frame (third frame), the STLBP histogram H over a circular region of radius R_{region} centered at the pixel (x, y) is computed using the last stored frame. Then, current histogram $H(x, y)$ is compared with the K model histograms using a proximity measure expressed in (16).

$$\bigcap(H_m, H) = \sum_{l=0}^{L-1} \min(H_{m,l}, H_l) \tag{16}$$

where $L = 2^l$ represents the number of bins. The intuition behind the histogram intersection in (16) is that it measures the common part of two histograms to measure the similarities between them.

For better understanding, let us take an example with $K = 4$ (multimodal Gaussian). Let the proximity measure between current histogram and four model histograms at pixel point (x, y) be \cap_1, \cap_2, \cap_3 and \cap_4 . Then updating the background model is same as in [26] as follows:

- If $\max(\cap_1, \cap_2, \cap_3, \cap_4) < T_p$ (T_p is the threshold value for proximity measure), then current histogram replaces the model histogram having lowest weight. The weight of this newly added histogram is initialized with a low value i.e. 0.01 for our experiment.
- If $\max(\cap_1, \cap_2, \cap_3, \cap_4) > T_p$, then we select the the best match model histogram having highest proximity value. The best matching model histogram with the current histogram is updated s follows:

$$H_{m_k}^{x,y} = \alpha_b \times H^{x,y} + (1 - \alpha_b) \times H_{m_k}^{x,y}, \quad \alpha_b \in [0, 1] \tag{17}$$

where α_b is an user defined learning rate.

The weights of the histogram is updated as follows:

$$\omega_k = \alpha_w \times M_k + (1 - \alpha_w) \times w_k, \quad \alpha_w \in [0, 1] \tag{18}$$

where α_w is an user defined learning rate, M_k is 1 for best matching model histogram and 0 for others.

In the suggested scheme, we have presented a newly adaptive updating learning rate (α_b) which is expressed in (19).

$$\alpha_b = C_1 \times \exp(C_2 \times (\max(..) - T_p)) \tag{19}$$

where, $\max(..)$ refers to the maximum proximity measure between models’ histogram with respect to the current. Again, C_1 and C_2 are constant values. The intuition behind this new adaptive updating learning rate is that if the matched model ‘ $\max(..)$ ’ is very close to the T_p (i.e. the difference ‘ $\max(..) - T_p$ ’ is very less), then the updating learning rate converges its lower value (i.e. C_1). When the matched model away in similarity from current (i.e. the difference ‘ $\max(..) - T_p$ ’ increases), the updating learning rate increases. This improves the convergence rate at each frame. It has been observed that, when background changes quickly under dynamic condition (unstable conditions), the matched model ‘ $\max(..)$ ’ very close to T_p . Therefore, a small learning rate is desirable for updating the model. On the contrary, if the background is stable (i.e. matched model ‘ $\max(..) \gg T_p$ ’ the model is updated with higher learning rate.

Further, in this work we have formulated to select an adaptive threshold value T_p which is dynamically changed with respect to each pixel as background also changes dynamically over time in a complex environment. Throughout this work we have considered $T_p = T_p(x, y)$. In the proposed scheme, we have used a counter which counts the the number of times the pixels identified as a foreground pixels. The counter value increases by one if the pixel at a particular location (x, y) along the temporal direction continuously classified as foreground pixel. If the pixel at (x, y) once classified as background in any t^{th} frame, counter is reset to zero. The criteria for aforementioned conditions are as follows:

$$\begin{cases} \text{Foreground} & count_t = count_{t-1} + 1 \\ \text{Background} & count_t = 0 \end{cases} \tag{20}$$

In illumination changes condition, some times it is noticed that the pixel is wrongly classified as foreground instead of background. However, in the succeeding frames the same location of the pixel is classified as background. This is due to the fact that scenario like a slight movement in leaf position may be classified as foreground whereas it is expected to be classified as background. Under this circumstances, the counter value is set to be zero to avoid such misclassification. If the counter value increases gradually, the pixel is considered as a foreground. During experimentation, we have assumed that if the counter value is above T_h^+ , the pixel is considered as foreground. It is expected that T_h^+ value should be maximum for foreground detection under challenging conditions. Once the counter value is above T_h^+ , the pixel is regarded as foreground and $T_p(x, y)$ value will be increased. The intuition behind to increase the $T_p(x, y)$ value is that, the pixel proximity value which are very close to $T_p(x, y)$ (i.e. $\max(..) \geq T_p(x, y)$) are classified as background. So to mitigate this situation, the $T_p(x, y)$ value is increased by a factor of $\frac{\beta}{count}$ amount. If the pixel at position (x, y) continuously detected as foreground the counter value increases which decreases the value of $\frac{\beta}{count}$. Therefore, T_p is approximately equal to T_{p-init} at the high value of counter additionally which leads to removal of false positive in a frame. The T_p value is maintained a constant for counter value within the range $[T_h^-, T_h^+]$. The same theory is adopted when $T_p(x, y)$ value decreases when the counter value is equal to zero. For better clarity, the proposed adaptive thresholding approach is outline in Algorithm 1. In Section 5.1, the influence of selecting an adaptive threshold $T_p(x, y)$ and learning rate α_b is presented in detail for better clarity.

Algorithm 1 Adaptive thresholding algorithm.

```

1 Begin
2 Input: Frames  $F_t$ 
3 Output: Detected binary image with adaptive thresholding approach
4 Step 1: Initialization of parameters. Initialize the counter  $count = 0$ , initial
    $T_p = T_{p-init} = 0.7, T_h^+ = 5, T_h^- = 3, \Delta = 0.02, \beta = 0.1$ 
5 if  $count \geq T_h^+$  then
6    $T_p = T_{p-init} + \frac{\beta}{count}$ 
7   else if  $count \geq T_h^-$  and  $count < T_h^+$  then
8      $T_p = T_{p-init}$ 
9   else
10     $T_p = T_{p-init} - \Delta$ 
11 End

```

All the produced model histograms are not certainly resulted by the background process. To determine whether the histogram models belong to background or not the persistency of the histogram in the model is computed. Equatino (18), shows that persistence is directly related to the model histogram’s weight. The larger the weight, the higher is the probability of being a background histogram produced by background process. During the final stage of the updating procedure, sort the model histograms at a pixel with decreasing order of their weights and select the most probable B background model histograms which are on the top of the list.

$$B = arg \min_b \left(\sum_{k=1}^b \omega_k > T_B \right), \quad T_B \in [0, 1] \tag{21}$$

where T_B is the user defined threshold.

For foreground detection, the histogram H at a pixel in the current frame is compared against the B background model histograms using proximity measure reflected in (16). This foreground detection process is done before the updation of the background model. For atleast one background model histogram, if the proximity measure is greater than T_p , the pixel is classified as background, otherwise the pixel is classified as foreground.

4.2 Parameters value selection in the proposed scheme

In the proposed scheme, the learning rate (α_b) and threshold value (T_p) are computed adaptively. However, there are some influence parameters which help to compute the above mentioned two parameters. A careful consideration must be applied to select the parameter values. There are so many tunable parameters used in the proposed scheme. So in most of the cases, constant value is adopted to reduce the overall computational burden and number of variables used in the algorithm. The contemplation for choosing the parameters value for different datasets are discussed as explained below.

Circular neighborhood LBP operator needs a proper value of I and R . If I value is more then the number of bin size (i.e. 2^I) which in turn increases the length of the histogram. On the other hand, if the value of I is considered to be small, there is a chance of information loss in the local region as the histogram length decreases. Furthermore, the memory size is an important factor for choosing the size of I . In the proposed scheme the radius size R is taken as 2. Selection of larger size of R decreases the correlation among the pixels, which causes the problem for finding the shape of the moving objects. Again, the smaller value of

R increases the number of false positives due to extraction of texture information which is too local. In this work, the value if I and R is taken experimentally as 6 and 2 respectively. The circular region of radius R_{region} refers to the region for histogram calculation. A large value of radius R_{region} incorporate large information into the histogram. Therefore, in a result, the shape of the moving objects are not extracted properly. However, the smaller value of radius R_{region} leads to proper detection of moving objects shape but creates large number of false positives. Therefore, the radius R_{region} is chosen as 5 to compromise both abovesaid requirements.

The requirements of number of models is distinct for different video sequences. It depends on the nature of complexity associated in each video. In the dynamic environment (i.e. outdoor scenario) which is a multimodal distribution, the value of K must be chosen high (preferably more than 5) and for an unimodal distribution environment (i.e. indoor), the value of K must be less (approximately 2). If higher value of K is taken, memory requirement and computational complexity increases. Further, for small value of K the detection result deteriorates to a great extent. In the suggested scheme, during experimentation for all videos, the value of K is considered as 4.

The selection parameter T_B directly related to the number of models required. The value of T_B should be very high (≈ 0.8) for multimodal background and small (≈ 0.2) for unimodal background. We have taken 0.8 for all our video sequences

The learning rate parameter (α_b, α_w) responsible for adaption speed of background model. In this work, only learning rate α_b is taken into consideration for updatation which is depicted in (19). Higher values of learning rate leads to faster updatation of current value histogram into the background model histogram. Faster updatation of background due to selection of high value of learning rate which causes the misclassification of slow moving object as background. On the contrary, low value of learning rate causes the slow updatation, which is responsible for increase in false positives. Therefore, α_b value in (19) changes in accordance with the speed of moving object in a video sequence. The value of learning parameter (α_w) is taken as 0.01 for all video sequences. To make a fair comparison, we have kept the parameters value constant for all datasets. Table 1 depicts the parameters value selection for all the datasets of the proposed scheme.

5 Experimental analysis

To appraise the efficaciousness of the proposed scheme, extensive simulations are carried out in MATLAB R2019a software running in a 64 bit windows with Intel Core i7 processor utilizing 32GB RAM under certain pertinent and specific test environment. To conduct the exhaustive simulations, some of the standard and widely accepted video sequences from different data sets are adopted with different challenging scenario. Further, the considered datasets are categorized into indoor and outdoor environment respectively. Table 2 depicts a detailed information about different video sequences used in the experiments. Further, the different and source of video sequences are listed in Tables 3 and 4 respectively. The

Table 1 The parameter values of the proposed method for all datasets

$LBPI_{I,R}$	R_{region}	T_r	K	$T_p-initial$	T_B	C_1	C_2	α_w	T_h^+	T_h^-	Δ	β	ω
$LBPI_{6,2}$	5	0.2	4	0.7	0.8	0.01	2	0.01	5	3	0.02	0.1	0.01

Table 2 Information about the video sequences used in the proposed scheme

Video	Scenario	Data set	Frame size	Description
Waving Tree (WT)	Outdoor	Wall flower	120×160	A person is walking with waving tree in background.
Campus (CA)	Outdoor	Perception.i2r	128×160	Vehicle and men are moving with waving of leaves at background.
Intelligent Room (IR)	Indoor	SBMnet	240×320	Man is walking in a room with constant illumination and high noisy environment.
Canoe (CAN)	Outdoor	Changedetection	240×320	Canoe is sailing with rippling of water surface swaying of vegetation at background
OTCBVS	Outdoor	OTCBVS	240×320	Men are moving with illumination changes condition because of sailing of clouds.
Highway (HW)	Outdoor	Changedetection	240×320	Vehicles are moving under small waving of leaves and different illumination condition at background
Fountain (FO1)	Outdoor	Perception.i2r	128×160	Men are moving with different speed with fountain at background.
Fountain (FO2)	Outdoor	Changedetection	288×432	Small vehicles are passing behind the fountain.
Watersurface (WS)	Outdoor	Perception.i2r	128×160	A person is walking and after-that standing at the bank of water surface.

Table 3 Challenges associated with different video sequences

Video sequences	Challenges in video sequences
WT	Dynamic background
CA	Dynamic background with Camouflage
IR	Constant illumination and noise environment
CAN	Rippling Water
OTCBVS	Illumination variation and shadow
HW	Dynamic background and illumination variation
FO1	Dynamic background and still position of moving object
FO2	Dynamic Background and small moving object
WS	Camouflage, rippling water and still position of moving object

Table 4 Download source of the video sequences used in our experiment

Video sequences	Download source
WT	https://www.microsoft.com/en-us/research/project/test-images-for-wallflower-paper/
CA, FO1, WS	http://vis-www.cs.umass.edu/~narayana/castanza/I2Rdataset/
IR	http://www.scenebackgroundmodeling.net/
CAN, FO2, HW	http://changedetection.net/
OTCBVS	[15]

proposed approach is compared with eight state-of-the-art approaches namely GMM [55], KDE [16], STBS [13], PBAS [27], T-BGS [26], STLBP-BGS [64], Goyal's method [20] and Moudgollya's method [41]. To retain about the fairness of the proposed scheme in entire experimentation and comparison no post processing operations are adopted to improve the overall detection performance of the proposed scheme. The overall experiment is segregated in terms of qualitative and quantitative evaluation measures.

In context to qualitative evaluation, visual assessment of frames for different video sequences are shown with respect to improved segmented foreground detection.

For quantitative measurement, performance matrices in terms of *Recall*, *Precision*, *F-measure*(F1), *Similarity*, percentage of correct classification (*PCC*) and average classification error (*ACE*) are computed. To give a better insights to the reader, following quantitative matrices are presented as follows:

$$Recall = \frac{TP}{TP + FN} \quad (22)$$

$$Precision = \frac{TP}{TP + FP} \quad (23)$$

$$F - measure = 2 \frac{Recall \times precision}{Recall + Precision} \quad (24)$$

$$PCC = \frac{TP + TN}{TP + TN + FP + FN} \quad (25)$$

$$Smilarity = \frac{TP}{TP + FP + FN} \quad (26)$$

$$ACE (AverageClassificationError) = \frac{FP + FN}{M} \quad (27)$$

where, M is the number of frames taken for evaluation purpose. The necessary description used for measuring the performance are represented as follows:

- I True Positive (TP): the number of pixels that are correctly detected as foreground pixels.
- II False Positive (FP):the number of background pixels that are incorrectly detected as foreground pixels.
- III True Negative (TN): the number of background pixels correctly marked as background pixels.
- IV False Negative (FN): the number of foreground pixels that are incorrectly marked as background pixels.

Additionally, the pseudo code to generate TP, TN, FP and FN over selected frames is described in Algorithm 2.

Algorithm 2 Algorithm for measurement of TP , TN , FP and FN .

```

1 Begin
2 Input: Detected binary frames  $BGS_t$  and ground truth  $G_t$ 
3 Output:  $TP$ ,  $TN$ ,  $FP$  and  $FN$ 
4 Step 1:  $N_{BGS}$ =Length of Detected Binary Frames
5 Step 2: Initialization of parameters. Initialize  $TP$ ,  $TN$ ,  $FP$  and  $FN$  set to 0
6 for  $t \leftarrow 1$  to  $N_{BGS}$  do
7   for each pixel  $(x, y)$  do
8     if  $BGS_t(x, y) = 1$  and  $G_t(x, y) = 1$  then
9        $TP = TP + 1$ 
10    else if  $BGS_t(x, y) = 0$  and  $G_t(x, y) = 1$  then
11       $FN = FN + 1$ 
12    else if  $BGS_t(x, y) = 0$  and  $G_t(x, y) = 0$  then
13       $TN = TN + 1$ 
14    else if  $BGS_t(x, y) = 1$  and  $G_t(x, y) = 0$  then
15       $FP = FP + 1$ 
16 End
    
```

5.1 Influence of adaptive threshold and learning rate

In this section, we show the influence of adaptive selection of threshold value (T_p) and learning rate (α_b) instead of choosing it heuristically. In this experiment, the qualitative as well as quantitative measures are evaluated using some video sequences namely WT and WS and shown in Figs. 8 and 9 respectively. In the proposed scheme i.e. STLBP (without

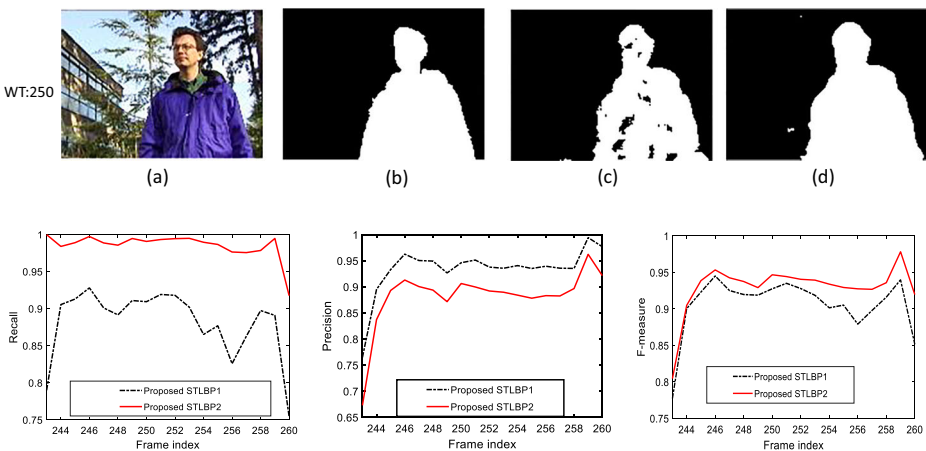


Fig. 8 Qualitative and Quantitative comparison of proposed STLBP1 and proposed STLBP2 . First row (Qualitative comparison): **a** 250th frame of WT video sequence, **b** Ground truth, **c** Proposed STLBP1 and **d** Proposed STLBP2 results. Second row shows the Quantitative comparison in terms of *Recall*, *Precision* and *F – measure* respectively of WT video sequences

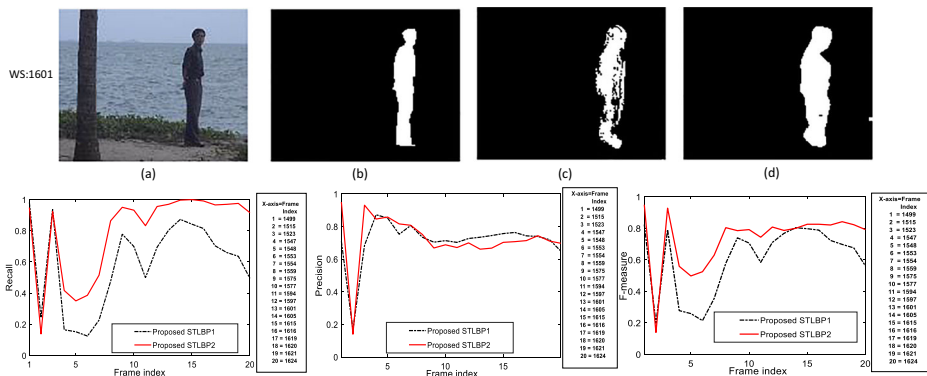


Fig. 9 Qualitative and Quantitative comparison of proposed STLBP1 and proposed STLBP2 on WS video sequence. First row (Qualitative comparison): **a** 1601st frame of WS video sequence, **b** Ground truth, **c** Proposed STLBP1 and **d** Proposed STLBP2 results. Second row shows the Quantitative comparison in terms of *Recall*, *Precision* and *F – measure* respectively of WS video sequences

adaptive parameter selection) the value of T_p and α_b are considered as 0.7 and 0.01 respectively. From the Figures it is clearly observed that the proposed scheme STLBP (with adaptive parameter selection) shows superior performance than that of the proposed scheme STLBP (without adaptive parameter selection). Henceforth, to make an uniformity for the subsequent results and analysis the proposed scheme without adaptive parameter selection and proposed scheme with adaptive parameter selection are named as proposed STLBP1 and proposed STLBP2 respectively.

5.2 Qualitative analysis

The qualitative comparison in terms of visual assessment of the proposed scheme (proposed STLBP2) than that of the benchmark schemes namely GMM [55], KDE [16], STBS [13], PBAS [27], T-BGS [26], STLBP-BGS [64], Goyal’s method [20] and Moudgollya’s method [41] using WT, HW, OTCBVS, CA, FO2, FO1, WS, IR and CAN video sequences are shown in Figs. 10, 11, 12, 13, 14 and 15. The video sequences used in the experiment constitute different complex situations like waving of tree, rippling water, low contrast, varying

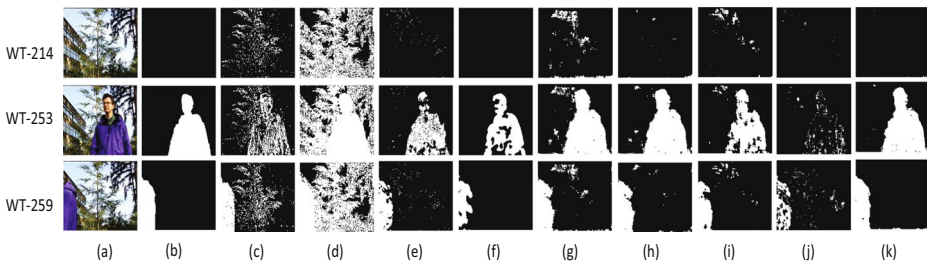


Fig. 10 Visual assessment of Moving object detection for WT video sequence: **a** original frames, **b** Ground Truth, **c** GMM, **d** KDE, **e** STBS, **f** PBAS, **g** T-BGS, **h** STLBP, **i** Goyal’s method, **j** Moudgollya’s method and **k** Proposed STLBP2

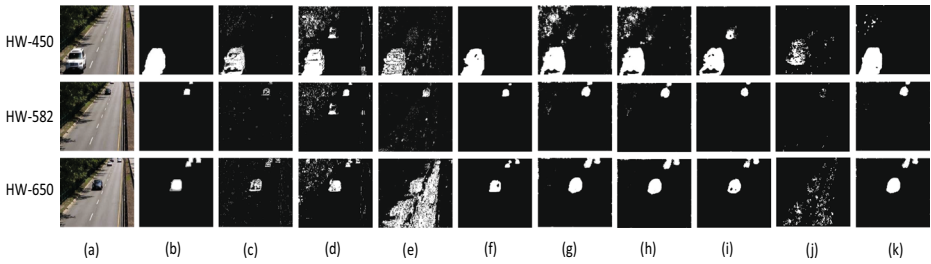


Fig. 11 Visual assessment of Moving object detection for HW video sequence: **a** Original frames, **b** Ground Truth, **c** GMM, **d** KDE, **e** STBS, **f** PBAS, **g** T-BGS, **h** STLBP, **i** Goyal’s method, **j** Moudgollya’s method and **k** Proposed STLBP2

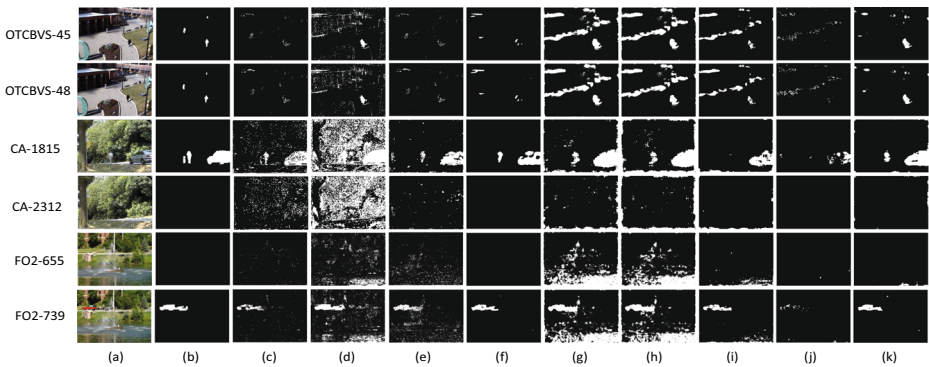


Fig. 12 Visual assessment of Moving object detection for OTCBVS, CA and FO2 videosequences: **a** Original frames, **b** Ground Truth, **c** GMM, **d** KDE, **e** STBS, **f** PBAS, **g** T-BGS, **h** STLBP, **i** Goyal’s method, **j** Moudgollya’s method and **k** Proposed STLBP2

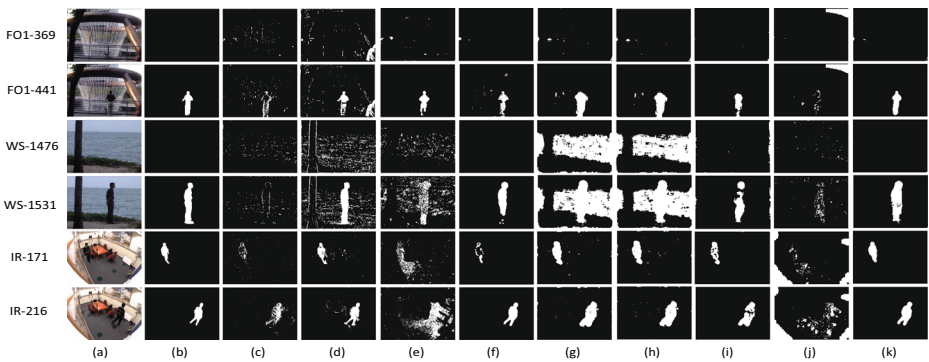


Fig. 13 Visual assessment of Moving object detection for FO1, WS and IR video sequences: **a** Original frames, **b** Ground Truth, **c** GMM, **d** KDE, **e** STBS, **f** PBAS, **g** T-BGS, **h** STLBP, **i** Goyal’s method, **j** Moudgollya’s method and **k** Proposed STLBP2

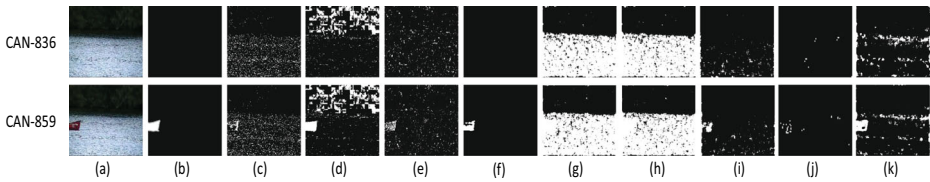


Fig. 14 Visual assessment of Moving object detection CAN video sequence: **a** Original frames, **b** Ground Truth, **c** GMM, **d** KDE, **e** STBS, **f** PBAS, **g** T-BGS, **h** STLBP, **i** Goyal's method, **j** Moudgollya's method and **k** Proposed STLBP2

illumination, camouflage, noisy and different speed of foreground object. Figure 10 shows the visual quality assessment comparison of the proposed scheme (proposed STLBP2) over that of the benchmark schemes. From the obtained result it is seen that the number of false

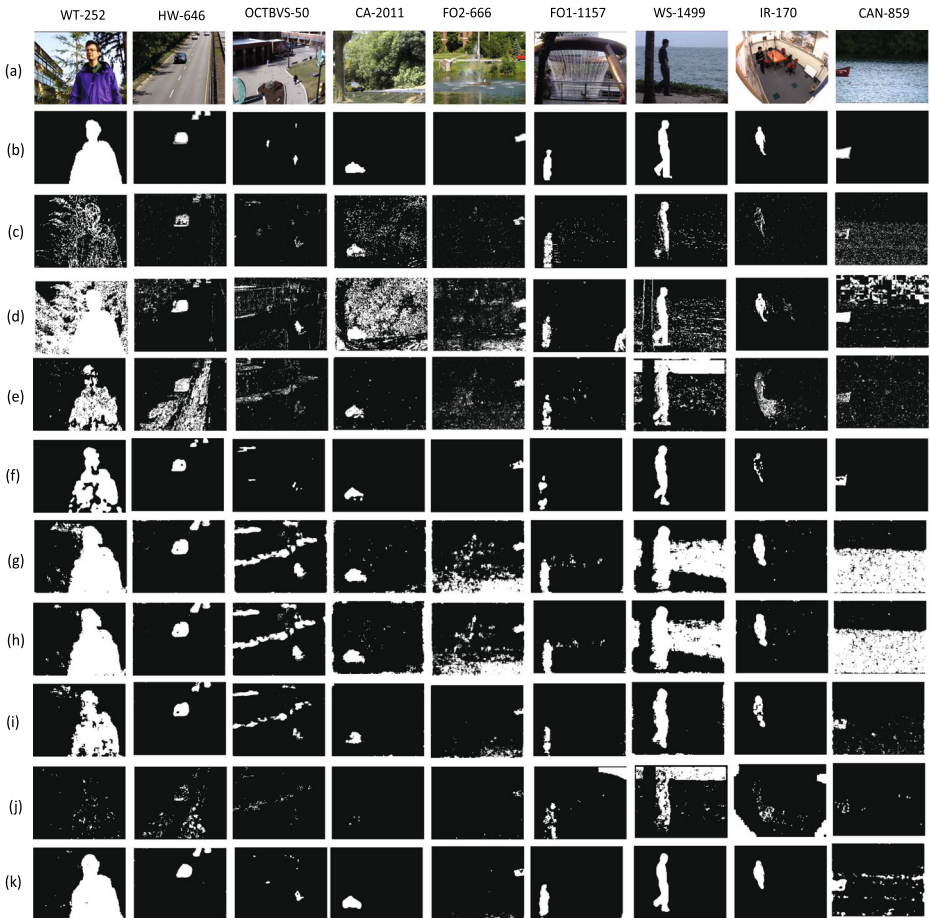


Fig. 15 Visual assessment of moving object detection for different video sequences: **a** Original frames, **b** Ground Truth, **c** GMM, **d** KDE, **e** STBS, **f** PBAS, **g** T-BGS, **h** STLBP, **i** Goyal's method, **j** Moudgollya's method and **k** Proposed STLBP2

positives and false negatives reduces significantly in the proposed STLBP2 scheme in comparison to benchmark schemes. In Fig. 11 for HW sequence, the proposed STLBP2 scheme brings potential improvement in terms of foreground extraction than that of PBAS, T-BGS, STLBP and Goyal's BGS approaches. However, techniques like GMM, KDE, STBS and Moudgollya's methods fails to extract the desired foreground with respect to ground truth.

Further, the visual assessment for some complex sequences like OTCBVS, FO2, and CA is presented in Fig. 12. These sequences exhibits complex scenario like change in illumination, rippling water, dynamic background with low contrast and camouflage. From Fig. 12, in-case of varying illumination (OTCBVS sequence) the proposed STLBP2 method exhibits illumination invariant nature and accurately detect the foreground object with reduced false positive rate. Further, for low contrast and dynamic sequences (CA and FO2) the proposed STLBP2 scheme shows similar improvement over state-of-the-art approaches. In particular, the proposed STLBP2 scheme minimizes the wrong classification and able to detect the foreground object accurately.

Figure 13 includes some complex sequences namely WS and FO1 where issues like dynamic background, ripple of water and noise are present. From the Figure it is noticed that for WS sequence the extracted foreground is demolished by background for T-BGS and STLBP method. This is due to the adoption of constant threshold value and learning rate. However, the proposed STLBP2 scheme is more visual appealing than that of the other schemes with reduced false detection. Similarly, for IR video sequence (a noisy sequence), the proposed STLBP2 scheme produces superior foreground detection in comparison to the competitive schemes.

Further, Fig. 14 contains most complex video sequence namely CAN where many challenging scenarios like illumination variation, rippling water, camouflage (i.e. some chromatic feature of background is similar as foreground) and still position of foreground object for a while are present. For CAN video sequence, the proposed STLBP2 scheme has not shown significant results for foreground detection as compared to BGS methods like KDE and PBAS.

A more detailed visual assessment representation of the proposed STLBP2 scheme over eight well accepted schemes is shown in Fig. 15. In conclusion, the proposed STLBP2 scheme shows impressive performance in most of the cases in terms of visual quality assessment than that of the reported schemes used in the experiments.

5.3 Quantitative analysis

The quantitative analysis and comparison of the proposed scheme (i.e. proposed STLBP1 and proposed STLBP2) is conducted in terms of *Recall*, *Precision*, *F-measure* and Percentage of correct classification (*PCC*), Average Classification Error (*ACE*) and *Similarity*.

Tables 5, 6, 7, 8, 9, 10, 11, 12 and 13, present the quantitative comparison of the proposed approach against eight different state-of-the-art BGS methods for different video sequences. From the obtained Tables 5–13 it is seen that the proposed STLBP2 scheme outperforms almost in all cases over other BGS methods in terms of *Recall*, *Precision*, *F-measure* and Percentage of correct classification (*PCC*), average classification error (*ACE*) and *Similarity*.

Figure 16a shows the average *Precision* for all BGS schemes. The average *Precision* value of PBAS is 0.7898 which is best among the methods. The next best average *Precision* value is 0.6866 obtained from the proposed STLBP2 method. The proposed STLBP2 method yields 13% less that of the highest reported scheme. Similarly, the average *Recall* value shown in Fig. 16b for T-BGS, STLBP, proposed STLBP1 and proposed STLBP2

Table 5 Quantitative Comparison of proposed method with that of other state-of-the-art BGS method for ‘WT’ data sequences

Algorithms	Metric					
	Precision	Recall	F_measure	PCC	Similarity	ACE
GMM	0.5049	0.7639	0.6080	87.98	0.4367	2305
KDE	0.1837	0.9997	0.3104	45.83	0.1837	10392
STBS	0.9160	0.7597	0.8305	96.22	0.7102	725
PBAS	0.6782	0.6380	0.6575	91.89	0.4897	1555
T-BGS	0.6866	0.9920	0.8115	94.38	0.6828	1078
STLBP	0.7500	0.9916	0.8540	95.87	0.7452	793
Goyal’s method	0.8218	0.8185	0.8201	95.62	0.6951	839
Moudgollya’s method	0.8689	0.4320	0.5771	92.28	0.4056	1481
Proposed STLBP1	0.8865	0.9841	0.9327	98.27	0.8740	332
Proposed STLBP2	0.9143	0.9976	0.9541	98.83	0.9122	225

Table 6 Quantitative Comparison of proposed method with that of other state-of-the-art BGS method for ‘HW’ data sequences

Algorithms	Metric					
	Precision	Recall	F_measure	PCC	Similarity	ACE
GMM	0.5982	0.5494	0.5728	98.38	0.4013	1227
KDE	0.3420	0.8725	0.4914	96.43	0.3257	2704
STBS	0.1138	0.6683	0.1945	92.96	0.1077	4423
PBAS	0.9774	0.9148	0.9451	99.79	0.8959	159
T-BGS	0.6649	0.9981	0.7982	99.03	0.6641	707
STLBP	0.6842	0.9978	0.8118	99.11	0.6832	650
Goyal’s method	0.8318	0.9674	0.8945	99.62	0.8091	287
Moudgollya’s method	0.0808	0.0988	0.0889	96.85	0.0465	590
Proposed STLBP1	0.7728	0.9888	0.8676	99.51	0.7661	355
Proposed STLBP2	0.9148	0.9988	0.9550	99.85	0.9139	113

Table 7 Quantitative Comparison of proposed method with that of other state-of-the-art BGS method for ‘OTCBVS’ data sequences

Algorithms	Metric					
	Precision	Recall	F_measure	PCC	Similarity	ACE
GMM	0.1655	0.1750	0.1706	98.87	0.0933	866
KDE	0.0790	0.7151	0.1423	94.28	0.0766	4385
STBS	0.0235	0.1358	0.0401	95.69	0.0205	3305
PBAS	0.1731	0.1188	0.1409	99.04	0.0758	737
T-BGS	0.0388	0.7100	0.0736	88.16	0.0382	9090
STLBP	0.0405	0.7198	0.0766	88.50	0.0399	8824
Goyal’s method	0.0395	0.3722	0.0714	93.58	0.0370	4927
Moudgollya’s method	0.0083	0.0254	0.0125	98.59	0.0063	268
Proposed STLBP1	0.1793	0.6471	0.2802	97.80	0.1633	1687
Proposed STLBP2	0.2085	0.7248	0.3238	98.20	0.1932	1375

Table 8 Quantitative Comparison of proposed method with that of other state-of-the-art BGS method for ‘CA’ data sequences

Algorithms	Metric					
	Precision	Recall	F_measure	PCC	Similarity	ACE
GMM	0.2358	0.7321	0.3565	93.44	0.2171	1343
KDE	0.0523	0.9938	0.0994	55.24	0.0523	9167
STBS	0.4350	0.6536	0.5224	97.93	0.3535	608
PBAS	0.7553	0.5270	0.6208	98.40	0.4501	327
T-BGS	0.2182	0.8709	0.3490	91.93	0.2114	1653
STLBP	0.1660	0.8686	0.2788	88.83	0.1620	2286
Goyal’s method	0.2401	0.1524	0.1864	96.70	0.1028	677
Moudgollya’s method	0.7600	0.0250	0.0484	97.87	0.0248	433
Proposed STLBP1	0.5234	0.8231	0.6399	97.70	0.4705	471
Proposed STLBP2	0.6513	0.9109	0.7593	98.57	0.6119	294

Table 9 Quantitative Comparison of proposed method with that of other state-of-the-art BGS method for ‘FO2’ data sequences

Algorithms	Metric					
	Precision	Recall	F_measure	PCC	Similarity	ACE
GMM	0.4754	0.5731	0.5197	98.55	0.3511	1796
KDE	0.1080	0.9793	0.1945	88.86	0.1077	13755
STBS	0.2194	0.8328	0.3473	96.97	0.2101	2922
PBAS	0.9628	0.8379	0.8960	99.73	0.8116	330
T-BGS	0.0815	0.9853	0.1506	83.70	0.0814	16576
STLBP	0.0864	0.9822	0.1588	84.82	0.0862	15568
Goyal’s method	0.3569	0.7942	0.4925	98.44	0.3267	1829
Moudgollya’s method	0.6054	0.1456	0.2348	99.04	0.1330	145
Proposed STLBP1	0.7024	0.8953	0.7872	99.50	0.6490	609
Proposed STLBP2	0.9123	0.9886	0.9489	99.78	0.9028	280

Table 10 Quantitative Comparison of proposed method with that of other state-of-the-art BGS method for ‘FO1’ data sequences

Algorithms	Metric					
	Precision	Recall	F_measure	PCC	Similarity	ACE
GMM	0.6150	0.4905	0.5457	97.17	0.3753	579
KDE	0.4710	0.7649	0.5830	96.21	0.4114	776
STBS	0.7583	0.6492	0.6995	98.07	0.5379	395
PBAS	0.8410	0.6272	0.7185	98.30	0.5607	348
T-BGS	0.5541	0.9359	0.6961	97.17	0.5339	579
STLBP	0.5655	0.9332	0.7042	97.29	0.5434	556
Goyal’s method	0.6929	0.4864	0.5716	97.48	0.4008	517
Moudgollya’s method	0.1431	0.2130	0.1712	94.42	0.0936	553
Proposed STLBP1	0.6656	0.8489	0.7461	98.00	0.5951	409
Proposed STLBP2	0.7202	0.8721	0.7889	98.38	0.6514	330

Table 11 Quantitative Comparison of proposed method with that of other state-of-the-art BGS method for ‘WS’ data sequences

Algorithms	Metric					
	Precision	Recall	F_measure	PCC	Similarity	ACE
GMM	0.4722	0.1658	0.2455	91.84	0.1399	1671
KDE	0.5055	0.9319	0.6555	92.16	0.4875	1606
STBS	0.3169	0.4826	0.3826	87.53	0.2365	2553
PBAS	0.8758	0.6880	0.7706	96.72	0.6269	671
T-BGS	0.1754	0.8567	0.2912	66.61	0.1704	6837
STLBP	0.1767	0.8217	0.2909	67.93	0.1702	6568
Goyal’s method	0.7040	0.7913	0.7451	95.67	0.5938	887
Moudgollya’s method	0.3643	0.1567	0.2191	91.06	0.1231	1830
Proposed STLBP1	0.6743	0.7943	0.7294	95.28	0.5741	966
Proposed STLBP2	0.7030	0.8611	0.7741	96.2	0.6314	760

Table 12 Quantitative Comparison of proposed method with that of other state-of-the-art BGS method for ‘IR’ data sequences

Algorithms	Metric					
	Precision	Recall	F_measure	PCC	Similarity	ACE
GMM	0.4820	0.3357	0.3957	98.70	0.2467	990
KDE	0.6409	0.9444	0.7636	99.26	0.6176	565
STBS	0.2184	0.5070	0.3053	98.27	0.1801	1174
PBAS	0.8475	0.3796	0.5243	99.13	0.3553	665
T-BGS	0.5337	0.9940	0.6945	98.96	0.5320	764
STLBP	0.5335	0.9924	0.6940	98.97	0.3313	756
Goyal’s method	0.6905	0.8655	0.7681	99.54	0.6236	349
Moudgollya’s method	0.0090	0.1149	0.0167	90.04	0.0084	1865
Proposed STLBP1	0.6193	0.9890	0.7617	99.29	0.6151	526
Proposed STLBP2	0.7975	0.9886	0.8829	99.70	0.7903	227

Table 13 Quantitative Comparison of proposed method with that of other state-of-the-art BGS method for ‘CAN’ data sequences

Algorithms	Metric					
	Precision	Recall	F_measure	PCC	Similarity	ACE
GMM	0.0290	0.5074	0.0548	92.54	0.0282	5714
KDE	0.0278	0.9997	0.0540	85.08	0.0278	11426
STBS	0.0795	0.5870	0.1400	97.79	0.0753	1254
PBAS	0.9945	0.8326	0.9064	99.83	0.8288	56
T-BGS	0.0106	0.9989	0.0209	51.27	0.0106	29803
STLBP	0.0111	0.9995	0.0220	52.52	0.0111	28698
Goyal’s method	0.2177	0.7186	0.3341	98.58	0.2006	978
Moudgollya’s method	0.1891	0.0235	0.0418	99.63	0.0213	70
Proposed STLBP1	0.1972	0.9552	0.3270	97.86	0.1954	1366
Proposed STLBP2	0.3331	0.9619	0.4948	99.35	0.3287	469

methods are 0.9269, 0.9230, 0.8806 and 0.9224 respectively. The average *F-measure* value is illustrated in Fig. 16c. However, it is noticed that the proposed STLBP2 scheme attains second and third best result for average *Precision* and average *Recall* value respectively.

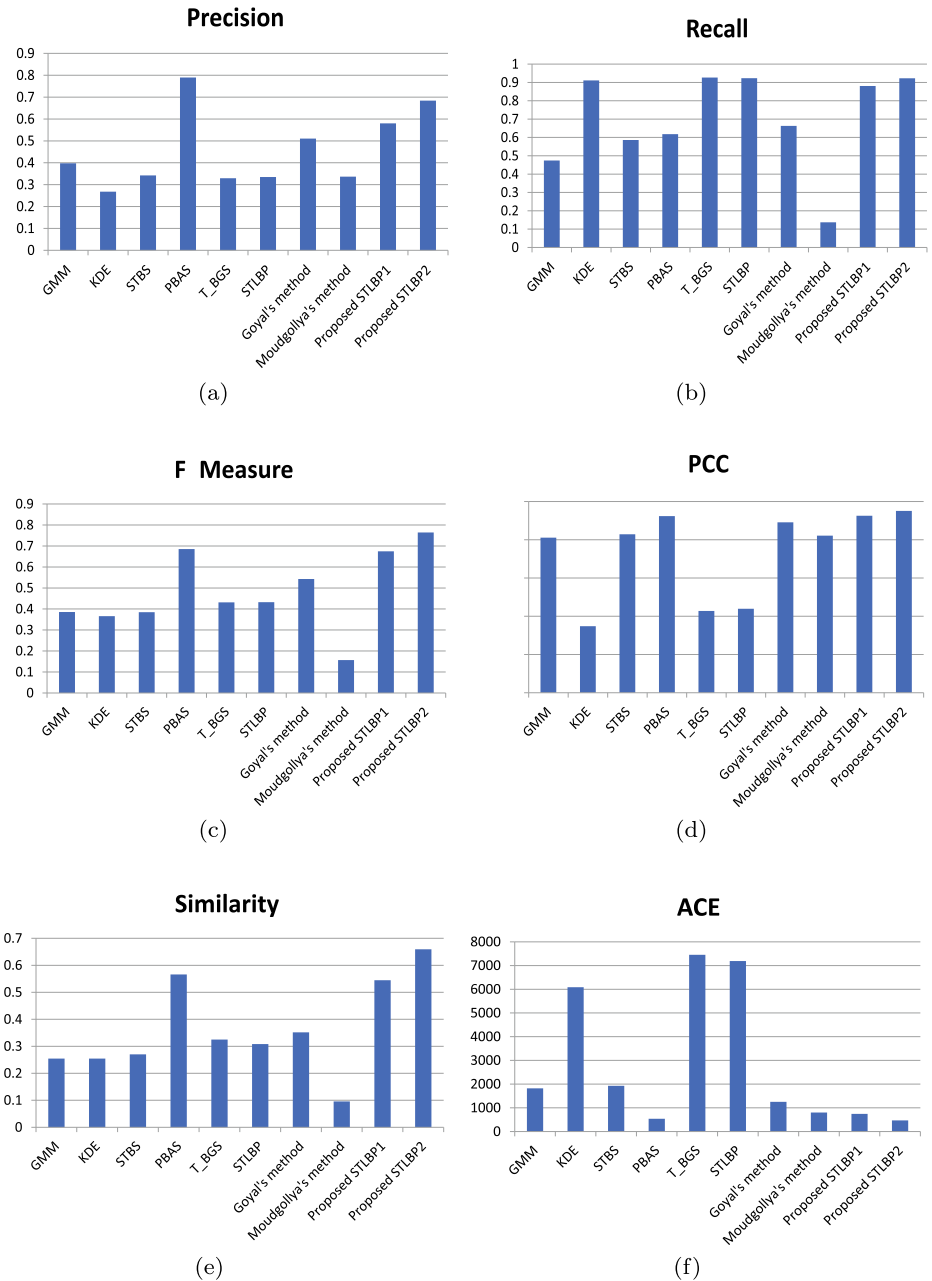


Fig. 16 Average Quantitative metrics comparison of different competing BGS methods with proposed method: **a** Precision, **b** Recall, **c** F-measure, **d** PCC, **e** Similarity, and **f** ACE

Table 14 Overall performance evaluation of algorithms

Cumulative score								
Algorithms	Precision	Recall	F-Measure	PCC	Similarity	ACE	Total Score	Rank
GMM	40	26	40	40	40	37	223	2
KDE	32	71	38	29	38	28	236	4
STBS	43	32	42	44	42	42	245	5
PBAS	80	33	64	75	68	73	393	8
T-BGS	26	77	37	25	36	23	224	3
STLBP	32	73	43	33	41	31	253	6
Goyal's method	60	37	54	57	54	56	318	7
Moudgollya's method	38	9	15	44	15	57	178	1
Proposed STLBP1	65	58	73	64	72	64	396	9
Proposed STLBP2	71	79	80	84	89	83	486	10

This is viable as the F-measure is the harmonic mean between *Precision* and *Recall*. From the result it is seen that the obtained average *F-measure* is the highest for the proposed STLBP2 scheme. The obtained average *F-measure* is 10.69% higher with respect to the second best method (i.e. PBAS). Similarly, for average *PCC* performance measures the proposed STLBP2 scheme and second best scheme (i.e. PBAS) achieved 98.74% and 98.09% respectively. The proposed scheme attains 0.66% higher value that of the second best scheme. Again, the improvements for the proposed STLBP2 approach is also noticed for average *Similarity* and average *ACE* as 17.27% and 13.28% respectively in comparison to the second best method depicted in Fig. 16.

Finally, overall performance [46] ranking between different BGS schemes based on the accuracy matrices is evaluated with respect to the proposed scheme and shown in Table 14. The ranking is made for ten BGS approaches using nine video sequences. The ranking strategy for evaluating the overall performance, the contenders are sorted in an ascending order of enumeration based on the performance (for *Precision*, *Recall*, *F-measure*, *PCC* and *Similarity*) and are assigned scores of 1, 2, 3, 4, ..., 10. However, for average classification error (*ACE*) the assigned values are arranged in a descending order. To conduct this experiment, a cumulative score is calculated by taking all considered test sequences. Next, the total score for a contender for different BGS algorithms is computed by adding its cumulative score for all performance matrices like *Precision*, *Recall*, *F-measure* etc. Finally the contenders are ranked as 10, 9, 8, ..., 1 (where 10 is the best and 1 is the worst) based on their overall total score. It is seen from the Table 14 that the proposed STLBP2 and proposed STLBP1 methods scored highest and second highest respectively in comparison to the other BGS schemes.

6 Conclusion

In the present work, an improved scheme for foreground detection is proposed under different complex scenarios. The proposed scheme initially utilized a spatio-temporal local binary pattern (STLBP) based approach to capture both spatial textural feature and temporal

motion feature from a video. In the suggested scheme, an improvement in change description rule of traditional STLBp method is made to capture robust features under challenging scenarios like waving tree, rippling water, camouflage, noisy video, low contrast and illumination variation. During background modeling step, instead of capturing a constant value for learning rate α_b and threshold value T_p , an adaptive formulation strategy is proposed to compute α_b and T_p to detect the foreground object more accurately with reduced false error rate. A thorough experimental analysis has been conducted on challenging video sequences. The performance of the suggested scheme has also been contrasted and validated in comparison to some benchmark schemes with similar experimental setup. From qualitative analysis and quantitative evaluations it is clearly noticed that the suggested scheme brings potential improvement in terms of accurate foreground detection with that of the state-of-the-art background subtraction approaches.

It is realized that the proposed algorithm is capable of detecting the foreground objects effectively under challenging conditions like illumination variation, noisy and dynamic background. The proposed BGS approach shows superior performance in terms of foreground detection under certain challenging conditions like dynamic background, illumination variation, shadow, low contrast and noisy sequences. However, the proposed scheme has not shown consistent results for foreground detection under camouflage and backgrounds with rippling of water conditions.

Developing an efficient foreground detection algorithm is still an open challenge in the field of computer vision. Future work should focus in the following directions: First, devise of multi-feature based foreground detection using different background conditions. Second, to formulate some alternate background modeling methods to enhance the foreground detection accuracy and further reduces the effect of false positive rate. Finally, exploring some recent deep learning techniques to segment the foreground accurately could be considered as another area of extension.

References

1. Ashraphijuo M, Aggarwal V, Wang X (2017) On deterministic sampling patterns for robust low-rank matrix completion. *IEEE Signal Process Lett* 25(3):343–347
2. Azizpour H, Laptev I (2012) Object detection using strongly-supervised deformable part models. In: *European conference on computer vision*. Springer, pp 836–849
3. Bilodeau GA, Jodoin JP, Saunier N (2013) Change detection in feature space using local binary similarity patterns. In: *2013 International conference on computer and robot vision (CRV)*. IEEE, pp 106–112
4. Bouwmans T (2011) Recent advanced statistical background modeling for foreground detection—a systematic survey. *Recent Pat Comput Sci* 4(3):147–176
5. Bouwmans T (2014) Traditional and recent approaches in background modeling for foreground detection: an overview. *Comput Sci Rev* 11:31–66
6. Bouwmans T, Garcia-Garcia B (2019) Background subtraction in real applications: challenges, current models and future directions. *arXiv preprint arXiv:190103577*
7. Bouwmans T, Zahzah EH (2014) Robust pca via principal component pursuit: a review for a comparative evaluation in video surveillance. *Comput Vis Image Underst* 122:22–34
8. Bouwmans T, El Baf F, Vachon B (2008) Background modeling using mixture of gaussians for foreground detection—a survey. *Recent Pat Comput Sci* 1(3):219–237
9. Bouwmans T, Porikli F, Höferlin B, Vacavant A (2014), Handbook on “background modeling and foreground detection for video surveillance”
10. Bouwmans T, Sobral A, Javed S, Jung SK, Zahzah EH (2017) Decomposition into low-rank plus additive matrices for background/foreground separation: a review for a comparative evaluation with a large-scale dataset. *Comput Sci Rev* 23:1–71
11. Chakraborty S, Paul M, Murshed M, Ali M (2017) Adaptive weighted non-parametric background model for efficient video coding. *Neurocomputing* 226:35–45

12. Chen BH, Huang SC (2014) An advanced moving object detection algorithm for automatic traffic monitoring in real-world limited bandwidth networks. *IEEE Trans Multimed* 16(3):837–847
13. Chiranjeevi P, Sengupta S (2012) Spatially correlated background subtraction, based on adaptive background maintenance. *J Vis Commun Image Represent* 23(6):948–957
14. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), IEEE vol 1, 886–893
15. Davis JW, Sharma V (2007) Background-subtraction using contour-based fusion of thermal and visible imagery. *Comput Vis Image Underst* 106(2–3):162–182
16. Elgammal A, Harwood D, Davis L (2000) Non-parametric model for background subtraction. In: European conference on computer vision. Springer, pp 751–767
17. Ershadi NY, Menéndez JM, Jimenez D (2018) Robust vehicle detection in different weather conditions: using mipm. *PloS One* 13(3) pp:1–30
18. Feng J, Xu H, Yan S (2013) Online robust pca via stochastic optimization. In: Advances in neural information processing systems, pp 404–412
19. Goehner K, Desell T, Eckroad R, Mohsenian L, Burr P, Caswell N, Andes A, Ellis-Felege S (2015) A comparison of background subtraction algorithms for detecting avian nesting events in uncontrolled outdoor video. In: 2015 IEEE 11th international conference on e-science. IEEE, pp 187–195
20. Goyal K, Singhai J (2018) Texture-based self-adaptive moving object detection technique for complex scenes. *Comput Electr Eng* 70:275–283
21. Goyette N, Jodoin PM, Porikli F, Konrad J, Ishwar P (2012) Changedetection. net: a new change detection benchmark dataset. In: 2012 IEEE computer society conference on computer vision and pattern recognition workshops. IEEE, pp 1–8
22. Hadi RA, George LE, Mohammed MJ (2017) A computationally economic novel approach for real-time moving multi-vehicle detection and tracking toward efficient traffic surveillance. *Arab J Sci Eng* 42(2):817–831
23. Hadiuzzaman M, Haque N, Rahman F, Hossain S, Siam MRK, Qiu TZ (2017) Pixel-based heterogeneous traffic measurement considering shadow and illumination variation. *Signal Image Video Process* 11(7):1245–1252
24. Han G, Wang J, Cai X (2017) Background subtraction based on modified online robust principal component analysis. *Int J Mach Learn Cybern* 8(6):1839–1852
25. Haritaoglu I, Harwood D, Davis LS (2000) W/sup 4: real-time surveillance of people and their activities. *IEEE Trans Pattern Anal Mach Intell* 22(8):809–830
26. Heikkila M, Pietikainen M (2006) A texture-based method for modeling the background and detecting moving objects. *IEEE Trans Pattern Anal Mach Intell* 28(4):657–662
27. Hofmann M, Tiefenbacher P, Rigoll G (2012) Background segmentation with feedback: the pixel-based adaptive segmenter. In: 2012 IEEE computer society conference on computer vision and pattern recognition workshops (CVPRW). IEEE, pp 38–43
28. Hong W, Kennedy A, Burgos-Artizzu XP, Zelikowsky M, Navonne SG, Perona P, Anderson DJ (2015) Automated measurement of mouse social behaviors using depth sensing, video tracking, and machine learning. *Proc Natl Acad Sci* 112(38):E5351–E5360
29. Hou YL, Pang GK (2011) People counting and human detection in a challenging situation. *IEEE Trans Syst Man Cybern-Part A: Syst Hum* 41(1):24–33
30. Huang SC (2011) An advanced motion detection algorithm with video quality analysis for video surveillance systems. *IEEE Trans Circ Syst Video Technol* 21(1):1–14
31. Huang SC, Chen BH (2013) Highly accurate moving object detection in variable bit rate video-based traffic monitoring systems. *IEEE Trans Neural Netw Learn Syst* 24(12):1920–1931
32. Javed S, Oh SH, Heo J, Jung SK (2014) Robust background subtraction via online robust pca using image decomposition. In: Proceedings of the 2014 conference on research in adaptive and convergent systems, pp 105–110
33. John F, Hipiny I, Ujir H (2019) Assessing performance of aerobic routines using background subtraction and intersected image region. In: 2019 International conference on computer and drone applications (IConDA). IEEE, pp 38–41
34. Karnowski J, Hutchins E, Johnson C (2015) Dolphin detection and tracking. In: 2015 IEEE Winter applications and computer vision workshops. IEEE, pp 51–56
35. Kim K, Khalidabhongse TH, Harwood D, Davis L (2004) Background modeling and subtraction by codebook construction. In: 2004 International conference on image processing, 2004. ICIP'04, vol 5. IEEE, pp 3061–3064
36. Lee B, Hedley M (2002) Background estimation for video surveillance. In: 2002 Image & Vision Computing New Zealand (IVCNZ'02), pp 315–320

37. Li Q, Bernal EA, Shreve M, Loce RP (2016) Scene-independent feature-and classifier-based vehicle headlight and shadow removal in video sequences. In: 2016 IEEE Winter applications of computer vision workshops (WACVW). IEEE, pp 1–8
38. Liu Y, Shi M, Zhao Q, Wang X (2019) Point in, box out: beyond counting persons in crowds. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 6469–6478
39. McFarlane NJ, Schofield CP (1995) Segmentation and tracking of piglets in images. *Mach Vis Appl* 8(3):187–193
40. Mei L, Guo J, Lu P, Liu Q, Teng F (2017) Inland ship detection based on dynamic group sparsity. In: 2017 ninth international conference on advanced computational intelligence (ICACI). IEEE, pp 1–6
41. Moudgollya R, Midya A, Sunaniya AK, Chakraborty J (2019) Dynamic background modeling using intensity and orientation distribution of video sequence. *Multimed Tools Appl* 78(16):22537–22554
42. Muniruzzaman S, Haque N, Rahman F, Siam M, Musabbir R, Hadiuzzaman M, Hossain S (2016) Deterministic algorithm for traffic detection in free-flow and congestion using video sensor. *J Built Environ Technol Eng* 1:111–130
43. Ojala T, Pietikäinen M, Harwood D (1996) A comparative study of texture measures with classification based on featured distributions. *Pattern Recognit* 29(1):51–59
44. Ojala T, Pietikäinen M, Maenpää T (2002) Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans Pattern Anal Mach Intell* 24(7):971–987
45. Panda DK, Meher S (2018) Adaptive spatio-temporal background subtraction using improved wronskian change detection scheme in gaussian mixture model framework. *IET Image Process* 12(10):1832–1843
46. Panda DK, Meher S (2018) A new wronskian change detection model based codebook background subtraction for visual surveillance applications. *J Vis Commun Image Represent* 56:52–72
47. Pang S, Zhao J, Hartill B, Sarrafzadeh A (2016) Modelling land water composition scene for maritime traffic surveillance. *Int J Appl Pattern Recognit* 3(4):324–350
48. Perrett T, Mirmehdi M, Dias E (2016) Visual monitoring of driver and passenger control panel interactions. *IEEE Trans Intell Transp Syst* 18(2):321–331
49. Piccardi M (2004) Background subtraction techniques: a review. In: 2004 IEEE international conference on systems, man and cybernetics, vol 4. IEEE, pp 3099–3104
50. Quesada J, Rodriguez P (2016) Automatic vehicle counting method based on principal component pursuit background modeling. In: 2016 IEEE international conference on image processing (ICIP). IEEE, pp 3822–3826
51. Roy A, Shinde S, Kang KD et al (2010) An approach for efficient real time moving object detection. In: ESA, pp 157–162
52. Shen W, Lin Y, Yu L, Xue F, Hong W (2018) Single channel circular sar moving target detection based on logarithm background subtraction algorithm. *Remote Sens* 10(5):742
53. Sobral A, Vacavant A (2014) A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. *Comput Vis Image Underst* 122:4–21
54. Sobral A, Bouwmans T, ZahZah EJ (2015) Double-constrained rpca based on saliency maps for foreground detection in automated maritime surveillance. In: 2015 12th IEEE international conference on advanced video and signal based surveillance (AVSS). IEEE, pp 1–6
55. Stauffer C, Grimson WEL (1999) Adaptive background mixture models for real-time tracking. In: IEEE computer society conference on computer vision and pattern recognition, 1999, vol 2. IEEE, pp 246–252
56. Tamás B (2016) Detecting and analyzing rowing motion in videos. In: BME scientific student conference, pp 1–29
57. Vaswani N, Bouwmans T, Javed S, Narayanamurthy P (2018) Robust subspace learning: robust pca, robust subspace tracking, and robust subspace recovery. *IEEE Signal Process Mag* 35(4):32–55
58. Wang H, Suter D (2005) A re-evaluation of mixture of gaussian background modeling [video signal processing applications]. In: IEEE international conference on acoustics, speech, and signal processing, 2005. Proceedings.(ICASSP'05), vol 2. IEEE, pp ii–1017
59. Wren CR, Azarbayejani A, Darrell T, Pentland AP (1997) Pfnder: real-time tracking of the human body. *IEEE Trans Pattern Anal Mach Intell* 19(7):780–785
60. Wu M, Peng X (2010) Spatio-temporal context for codebook-based dynamic background subtraction. *AEU-Int J Electr Commun* 64(8):739–747
61. Yang B, Zou L (2015) Robust foreground detection using block-based rpca. *Optik* 126(23):4586–4590
62. Yang D, Zhao C, Zhang X, Huang S (2017) Background modeling by stability of adaptive features in complex scenes. *IEEE Trans Image Process* 27(3):1112–1125
63. Yousif H, Yuan J, Kays R, He Z (2017) Fast human-animal detection from highly cluttered camera-trap images using joint background modeling and deep learning classification. In: 2017 IEEE international symposium on circuits and systems (ISCAS). IEEE, pp 1–4

64. Zhang S, Yao H, Liu S (2008) Dynamic background modeling and subtraction using spatio-temporal local binary patterns. In: 15th IEEE international conference on image processing, 2008. ICIP 2008. IEEE, pp 1556–1559
65. Zhong B, Liu S, Yao H, Zhang B (2009) Multi-resolution background subtraction for dynamic scenes. In: 2009 16th IEEE international conference on image processing (ICIP). IEEE, pp 3193–3196
66. Zhong Z, Zhang B, Lu G, Zhao Y, Xu Y (2017) An adaptive background modeling method for foreground segmentation. *IEEE Trans Intell Transp Syst* 18(5):1109–1121

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.