

**A texture descriptor: BackGround Local Binary Pattern (BGLBP)**

**Author**

Hashem Davarpanah, S, Khalid, Fatimah, Abdullah, Lili Nurliyana, Golchin, Maryam

**Published**

2015

**Journal Title**

Multimedia Tools and Applications

**Version**

Accepted Manuscript (AM)

**DOI**

<https://doi.org/10.1007/s11042-015-2588-3>

**Copyright Statement**

© 2015 Springer. This is an electronic version of an article published in American Journal of Clinical Dermatology, June 2016, Volume 75, Issue 11, pp 6549–6568. American Journal of Clinical Dermatology is available online at: <http://link.springer.com/> with the open URL of your article.

**Downloaded from**

<http://hdl.handle.net/10072/101354>

**Griffith Research Online**

<https://research-repository.griffith.edu.au>

# A texture descriptor: BackGround Local Binary Pattern (BGLBP)

S. Hashem Davarpanah<sup>1</sup> · Fatimah Khalid<sup>2</sup> ·  
Lili Nurliyana Abdullah<sup>2</sup> · Maryam Golchin<sup>3</sup>

Received: 26 August 2014 / Revised: 11 February 2015 / Accepted: 30 March 2015  
© Springer Science+Business Media New York 2015

**Abstract** Local Binary Pattern (LBP) is invariant to the monotonic changes in the grey scale domain. This property enables LBP to present a texture descriptor being useful in applications dealing with the local illumination changes. However, the existing versions of LBP are not able to handle image illumination changes, especially in outdoor environments. The non-patterned illumination changes disturb performance of the background extraction methods. In this paper, an extended version of LBP which is called BackGround LBP (BGLBP) is presented. BGLBP is designed for the background extraction application but it is extendable to the other areas as a texture descriptor. BGLBP is an extension of D-LBP, Centre-Symmetric LBP, ULBP, and R-LBP and it has been designed to inherit the positive properties of previous versions. The performance of BGLBP as a part of background extraction method is investigated. In addition, a comparison between BGLBP as a general texture descriptor and a number of LBP versions is conducted.

**Keywords** LBP · Texture descriptor · Background extraction · Moving detection

---

✉ S. Hashem Davarpanah  
davarpanah@usc.ac.ir

Fatimah Khalid  
fatimahk@upm.edu.my

Lili Nurliyana Abdullah  
liyana@upm.edu.my

Maryam Golchin  
maryam.golchin@griffithuni.edu.au

<sup>1</sup> Department of Computer Engineering, Faculty of Engineering, University of Science and Culture, Tehran, Iran

<sup>2</sup> Faculty of Computer Science and IT, Universiti Putra Malaysia, Serdang 43300, Malaysia

<sup>3</sup> School of ICT, Griffith University, Gold Coast Campus, Southport, QLD 4222, Australia

## 1 Introduction

LBP can be used in the background extraction process. The intensity of an image is multiplication of the illumination and the reflectance. The illumination changes cause unpredictable intensity values and result in an unexpected image difference in the background subtraction process. This occurrence interferes with the accuracy of the motion detection process [21]. Recently, Local Binary Pattern (LBP) has been used to describe pixel properties. The basic principle of LBP, as a texture descriptor, is a scene being seen as a composition of micro-patterns generated by a concatenation of circular binary gradients [28]. The histogram of these illumination invariant micro-patterns is used as a behavior descriptor of the corresponding pixel, block or region.

LBP is invariant to monotonic changes in the grey scale domain [10]. An LBP histogram does not present information about the positions where the individual LBP codes have been calculated. These properties make LBP histograms robust against illumination changes. LBP histograms also support multi-modal backgrounds. They are computationally fast. Furthermore, more than one LBP histogram can be used to model each block [10].

Our experiments show that manipulating the definition of LBP and its calculation method increases its description power to describe the changes of pixel intensity, especially when illumination changes are large and take place quickly. These experiments led us to introduce an extended version of LBP called BGLBP, which stands for BackGround Local Binary Pattern. Due to the characteristics of BGLBP, LBP is therefore effective in other applications as a powerful texture descriptor.

In this paper, the performance of BGLBP is investigated in three different aspects. First, a comparison between BGLBP and a number of other versions of LBP is undertaken. Next, the effect of BGLBP is evaluated on the background extraction application. Finally, the effectiveness of BGLBP as a texture classifier is tested.

## 2 Local binary pattern

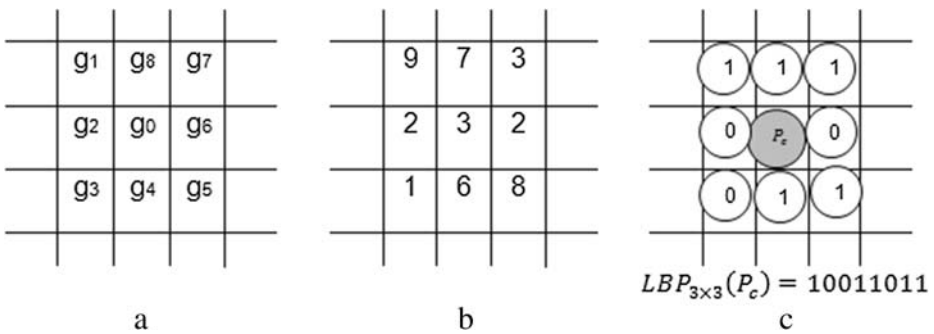
LBP was first introduced in 1994 [18]. A local binary pattern operator offers an effective way of analyzing textures. LBP is simple and combines the properties of structural and statistical texture analysis methods. It is invariant to monotonic gray-scale changes. The original version of LBP is calculated according to the following formula:

$$LBP_{R,N}(x, y) = \sum_{i=1}^{N-1} s(p_i - p_c) \times 2^i \quad (1)$$

where  $p_c$  and  $p_i$  stand for gray values of the central pixel, and each neighbor pixel of  $N$  equally spaced pixels on a circular path of radius  $R$ , respectively. The function  $s(x)$  is defined as follows:

$$s(x) = \begin{cases} 1 & x \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Original LBP considers a  $3 \times 3$  neighborhood as shown in Fig. 1. In this picture, eight neighbor pixels of the central pixel are determined in Fig. 1a; an example of pixel values is given in Fig. 1b; and the LBP for the given sample is calculated in Fig. 1c.



**Fig. 1** a A  $3 \times 3$  neighborhood of  $g_0$  b the pixels' intensity values c the Binary number assigned to  $p_i$  starting from top-left anticlockwise. Picture from [1]

LBP can be calculated for each block separately. In Dong, 2010 [22], to calculate the LBP value of each block (Multi-Block LBP), an operator is defined by comparing the average intensity of the central block with its neighbor blocks. In [17], an edge histogram is used to calculate the LBP value of each block. Using pixel-specific edge direction, bin value of histograms is constructed and is increased using the edge magnitudes. In addition, the LBP histogram of each block is calculated based on the LBP values of pixels belonging to the block. This histogram can be used as a texture descriptor of the corresponding block [25].

The Center-Symmetric Local Binary Pattern (CS-LBP), which was first used for matching and object-category classification is an effective extension of LBP [9]. This operator is illumination invariant, more robust to noise, and it produces short histograms. Gengjian et al. [25] introduced an operator named Spatial Center-Symmetric Local Binary Pattern (SCS-LBP), which is special for pixel-based background modeling. Another improved version of CS-LBP is presented by [24]. This operator classifies the local pattern based on relativity of the central pixel and the center-symmetric pixels instead of the gray value differences between the center-symmetric pixels. The authors named their method D-LBP (Direction Local Binary Pattern). No thresholds are needed in D-LBP; it is difficult to choose the adaptable threshold in CS-LBP [12].

Uniform LBP (ULBP) is an extension of the original LBP. ULBP reduces the feature dimensions and increases the noise immunity. Uniform LBPs include most of the LBPs and hence all the non-uniform LBPs are usually gathered into one single bin of the LBP diagram. This feature reduces the number of histogram bins and enhances the classification efficiency [7]. A uniform LBP is an LBP that contains at least two bitwise 0–1 or 1–0 transitions. L. Zhang et al. [27] extended the original LBP feature and proposed the Multi-Block LBP (MB-LBP), which is motivated by using Harr-like features.

T. Ojala et al. [19] proposed the Rotation Invariant Uniform Local Binary Pattern (RIU-LBP), an operator by discarding the direction of the information which results a low-dimensional feature. Local Ternary Pattern (LTP), a ternary version of LBP, is more resistant to noise. Wen-Hung et al. [15] defined Uniform LTP. Yonggang et al. [8] introduced Gradient Local Binary Pattern (GLBP). GLBP increases the performance of conventional local binary pattern methods when the reflected illumination changes rapidly. Volume Local Binary Pattern (VLBP) was introduced by Guoying et al. [29]. VLBP is based on using temporal information to calculate the LBP value for each pixel.

### 3 BGLBP: a new version of local binary pattern

The summary of each version of LBP is listed in Table 1. Two critical properties are required for a desired LBP version in the background extraction. 1) It should be fast and the number of bins in the plotted histogram should be the least, and 2) It should be calculated based on whole pixels' values belonging to each block. Therefore, the original LBP, LTP, and the Gradient LBP are not recommended. The Multi Block LBP calculates a LBP value for each block instead of each pixel separately. It is calculated exactly based on the minimum, average, maximum or an approximate value of the pixels belonging to the corresponding block. Working on these derived values reduces the sensitivity to the level of illumination changes and this version of LBP is not recommended for background extraction. ULBP is not rotation invariant and finally there are many bins in the Center-Symmetric LBP model. Regarding all of these limitations, BGLBP is introduced in this paper in order to overcome these problems.

BGLBP is calculated according to the following equations:

$$BGLBP_{P,R} = \begin{cases} \sum_{i=0}^{\left(\frac{P}{2}\right)-1} s\left(g_i, m, g_{\left(\frac{P}{2}\right)+i}\right) \times 2^i & U\left(LB\frac{P}{2}_{P,R}\right) \leq 2 \\ 2_2^P & otherwise \end{cases} \quad (3)$$

$$s\left(g_i, m, g_{\left(\frac{P}{2}\right)+i}\right) = \begin{cases} 1 & \text{if } \left( \left( g_i \geq m \geq g_{\left(\frac{P}{2}\right)+i} \right) \vee \left( g_i < m < g_{\left(\frac{P}{2}\right)+i} \right) \right) \&\& \left( \left( abs(g_i - m) + abs(g_{\left(\frac{P}{2}\right)+i} - m) \right) \geq \beta \right) \\ 0 & otherwise \end{cases} \quad (4)$$

$$m = \frac{1}{P} \left( g_c + \sum_{i=1}^{P-1} g_i \right) \quad (5)$$

**Table 1** Summary of LBP versions

LBP Type	Properties
Original LBP [18]	<ul style="list-style-type: none"> <li>• Invariant to monotonic changes in gray scale.</li> <li>• Computationally fast</li> </ul>
Center-Symmetric LBP [9, 10]	<ul style="list-style-type: none"> <li>• Faster than LBP</li> <li>• This version builds short histograms</li> <li>• More robust to noise</li> <li>• Illumination invariance</li> </ul>
D-LBP [12]	<ul style="list-style-type: none"> <li>• Extension of CS-LBP</li> <li>• No threshold needed</li> </ul>
Uniform LBP [7, 26]	<ul style="list-style-type: none"> <li>• Reduced number of bins in the histogram (non-uniform LBPs are gathered into one single bin)</li> <li>• Increases noise immunity</li> </ul>
Multi block LBP [22, 27]	<ul style="list-style-type: none"> <li>• Calculates LBP for each block instead of each pixel</li> </ul>
Rotation invariant LBP [19]	<ul style="list-style-type: none"> <li>• Local rotation invariant</li> </ul>
Local ternary patterns [14, 15]	<ul style="list-style-type: none"> <li>• Less sensitive to the noise</li> <li>• Complexity is increased</li> </ul>
Gradient LBP [8]	<ul style="list-style-type: none"> <li>• More accuracy in sudden illumination changes</li> <li>• More complex</li> </ul>

$g_i (i=0, \dots, i=P-1)$  corresponds to the gray values of  $P$  equally spaced pixels on a circular path of radius  $R$  that forms a circular symmetric neighbor set.  $g_c$  presents the intensity value of the central pixel,  $U$  refers to a uniform pattern.

In this formula, first, the average of local intensity values is calculated (Eq. (5)). The average value is used as an intensity threshold value to detect the changes in the neighbor pixels. This is used to reduce the noise. Equation (4) detects the intensity changes of  $i$ th pixel when the pixel's intensity value is more than the average value and the average value is more than intensity value of the corresponding diagonal pixel, or when the pixel's intensity value is less than the average value and the average value is less than the intensity value of the corresponding diagonal pixel. Furthermore, the difference between these changes is more than  $\beta$ . Using the earlier condition, the small illumination changes in the image are disregarded. This helps to detect large changes more appropriately. Since the summation of two different values is utilized with the same minimum values, the difference is regarded as an illumination change in the BGLBP. Using Eq. (3), half the changes of the uniform neighbor pixels are accounted. A pattern is uniform if the number of the corresponding changed bits is less than or equal to 2.

### 3.1 Advantages of BGLBP

BGLBP gains the advantages of other LBP versions.

The time and memory complexity of BGLBP is less than the LBP. Since BGLBP is based on ULBP, the number of bins is much less than the number of bins in the LBP ( $\log_2 n+1$  vs.  $n$ ). BGLBP needs less memory and less time to calculate the histogram. However, computing the number of Uniform LBPs is time consuming. Overall, the time complexity of BGLBP is less but its time consumption is not.

- BGLBP is less noise sensitive. The BGLBP histogram is calculated for each block. The noise which is applied to a number of pixels in the block hardly affects the accuracy of the decision, which is made based on the calculated BGLBP histogram. In addition, to calculate the BGLBP for each pixel, average of the intensity values of the corresponding pixel and its neighbor pixels is calculated and used. Using the average value makes the BGLBP less sensitive to noise. Moreover, small changes are not considered in the BGLBP so that small noises are omitted.
- To calculate the BGLBP histogram, the mean value of the background Gaussian model (temporal information) and the spatial information is used. Using temporal information in BGLBP causes sensitivity to any occurring gradual illumination changes in each pixel and in each block from time to time.
- Updated BGLBP histogram is more sensitive to large intensity changes.
- BGLBP can disregard the small illumination changes in the image and help to detect large changes more appropriately. At the first view, combination of these two opposite properties is not possible. Using blocking makes able background extraction method to neglect small changes. A block to be detected as a moving block should have a minimum number of pixels with different illumination values; a small number of changed pixels is less than the required minimum threshold value thus small changes are neglected. On the other hand, in calculation of the BGLBP histogram for each block, Eq. 6, big illumination changes occurring in the sudden changes, for example when a vehicle with completely different gray values is entered to the scene, are counted two times. It means blocks including pixels with big different gray values are detected as the moving blocks. Therefore the algorithm is sensitive to the sudden illumination changes.

In addition, the histogram calculation method is changed in order to apply the effect of large changes in the intensity values of each pixel in any block. If the difference between the average intensity values of the neighbor pixels and its corresponding mean value of the background Gaussian model is larger than a predefined threshold, the corresponding bin of the calculated BGLBP is increased by 2, otherwise it is increased by 1. This technique causes the BGLBP to be more robust against unexpected illumination changes.

$$Bin_i^{BGLBP} += \begin{cases} 2 & |m - \mu_{g_c}| \geq T \\ 1 & otherwise \end{cases} \quad (6)$$

where  $\mu_{g_c}$  stands for the mean value of the background Gaussian model in the corresponding pixel.  $T$  is a threshold which is calculated experimentally.

#### 4 Influence of the parameters

Before discussing the parameters and their best influences, it is required to introduce the performance metrics. In all our experiments, the following performance metrics were used:

- Execution time of the methods to compare corresponding time consumptions.
- Root Mean Square error (RMS) is calculated using Eq. (7).

$$\theta = \frac{1}{M \times N} \sum_{x=1}^M \sum_{y=1}^N (BG(x,y) - GT(x,y))^2 \quad (7)$$

where,  $M$  and  $N$  are dimensions of the current image;  $BG$  refers to the extracted background image, and  $GT$  denotes the Ground Truth image.

- It is possible to calculate the mean distance of the patterns belonging to a cluster because of the k-way partitioning [15]. The mean distance is regarded as an indicator of cluster homogeneity. This factor is calculated according to the Eq. (8).

$$X^2(S, T) = \sum_i \frac{(BG(i) - GT(i))^2}{(BG(i) + GT(i))} \quad (8)$$

- The False Detection Rate (FDR) is the fourth quantitative factor.
- Finally, the Peak Signal to Noise Ratio (PSNR) is calculated and defined as Eq. (9):

$$PSNR = 10 \log \left( \frac{\sum_{i=1}^M \sum_{j=1}^N (BG(i,j))^2}{\sum_{i=1}^M \sum_{j=1}^N (GT(i,j) - BG(i,j))^2} \right) \quad (9)$$

where  $GT$  refers the standard background image and  $BG$  stands for the background image extracted by the method. In addition,  $M$  and  $N$  denote the size of the image. Since PSNR explains peak of the signal to the noise, a larger value of this parameter is more acceptable.

**Table 2** Impact of different min distance threshold values using the Shopping Mall (SM) dataset

Minimum distance threshold	Time (ms)	FDR (pixels)	$\chi^2$	PSNR	RMS (pixels)
1	<b>72.12</b>	21180	26465	20.7284	<b>161,172,246</b>
2	74.21	21174	26452	20.7297	161,151,259
3	76.76	<b>21165</b>	26433	<b>20.7309</b>	161,127,310
4	76.877	21169	26423	20.7309	161,125,389
5	76.70	21172	<b>26418</b>	20.7307	161,128,531

The values in bold are the best results in each column

#### 4.1 Distance threshold parameter ( $\beta$ )

The threshold ( $\beta$  in Eq. 4) is set to 3 from the experiments shown in Table 2. This table explains the impact of minimum distance difference values on the accuracy of the extracted background image.

#### 4.2 Threshold parameter ( $T$ )

In order to estimate the  $T$  threshold (in Eq. 6), the Shopping Mall (SM) as a crowded indoor environment and Pets2000 (P0) as a semi static outdoor scene were selected and different threshold values in two image sequences were investigated. For each dataset, an image from the dataset including no object was manually selected as the ground truth image. The results were shown in Tables 3 and 4. In these two tables, the desired factors were calculated using different values of  $T$ . A range between 0 and 255 has been assigned to the threshold and the best results were achieved between 40 and 80. Finally 75 was assigned to the threshold value. This value was constant in the rest of the image sequences.

### 5 Analysis of BGLBP

The assessment of BGLBP was performed from three different viewpoints as indicated below. First, a comparison between BGLBP and a number of other versions of LBP was undertaken. Furthermore, the effect of BGLBP was evaluated on the background extraction application. Finally, the effectiveness of BGLBP as a texture classifier was tested.

**Table 3** Impact of different values of the  $T$  parameter using the SM dataset

T	Time (ms)	FDR (pixels)	$\chi^2$	PSNR	RMS (pixels)
40	77.79	21,190	26,422	20.7291	161,183,338
50	76.48	21,175	<b>26,416</b>	20.7303	161,152,620
60	75.63	21,177	26,419	20.7306	161,141,863
65	<b>75.60</b>	21,172	26,420	20.731	161,128,332
70	76.89	21,170	26,430	20.731	<b>161,126,955</b>
75	75.82	<b>21,165</b>	26,433	<b>20.731</b>	161,127,310
80	75.65	21,170	26,443	20.7303	161,128,417
90	75.81	21,172	26,449	20.7297	161,141,734
100	75.17	21,180	26,454	20.729	161,163,767

The values in bold are the best results in each column



**Table 4** Impact of different values of the T parameter using the PETS2000 (P0) dataset

T	Time (ms)	FDR (pixels)	$\chi^2$	PSNR	RMS (pixels)
40	108.32	5,502	89,224	19.5566	114,029,939
50	107.87	5,502	89,228	19.5565	114,036,089
60	107.88	5,503	89,228	19.5566	114,029,464
65	106.78	5,503	89,233	19.5566	114,028,934
70	106.65	5,503	89,233	19.5566	114,027,994
75	<b>98.09</b>	<b>5,501</b>	<b>89,211</b>	<b>19.5581</b>	114,019,061
80	107.72	5,501	89,228	19.5567	114,025,002
90	106.64	5,503	89,236	19.5568	<b>114,014,062</b>
100	106.63	5,502	89,238	19.5567	114,017,954

The values in bold are the best results in each column

### 5.1 Analysis of BGLBP vs. other versions of LBP in background image extraction

As it was mentioned before, BGLBP is used as a part of a background extraction method; it has been introduced to enhance the sensitivity of LBP when it faces with intensity changes. In this section, the performance of the BGLBP as a part of a background extraction method is investigated in comparison with the other versions of LBP.

Among the numerous versions of LBP, the following versions of LBP were selected and re-implemented in this research, namely, Original LBP, Circular LBP, Uniform LBP, ICS-LBP (Improved Center Symmetric LBP) [24], ID-LBP (Improved Direction LBP) [12], RIU-LBP (Rotation Invariant Uniform LBP) [16], MB-LBP (Multi-block LBP), and CMB-LBP (Circular Multi-Block LBP). The methods were analyzed using the PET2000 (P0) [20] and SM [11] image sequences. These two datasets were selected as sparse and crowded scenes, respectively. For the background extraction method, the proposed method in [5] was used. The experiments were done with and without the initialization process and the results were presented in Tables 5 and 6 respectively. The results showed that the desired factors have more acceptable values after applying BGLBP compared to other LBP versions. If the initializing method was also applied, the results of the background extraction method including BGLBP would be more accurate.

It is clear that the improvement of the results is not significant. This is due to two reasons: first, desired factors are obtained by averaging over a number of blocks. These blocks belong to a sequence of frames. Events such as sudden changes in the intensity, rotation, entrance of

**Table 5** Performance of each LBP version using the SM Dataset with initialization

Method	Time (ms)	FDR (pixels)	$\chi^2$	PSNR	RMS (pixels)
Original LBP	70.83	21220	26349	20.7299	161,260,665
MB Block	58.40	21228	26468	20.7235	161,279,805
CMB Block	<b>40.87</b>	21250	26575	20.7171	161,392,293
ULPB	70.87	21219	26372	20.7285	161,256,899
ICS-LBP	63.58	21210	26379	20.73	161,210,903
ID-LBP	71.52	21212	26365	20.7305	161,214,102
RIU-LBP	74.15	21208	26398	20.7278	161,221,927
Circular LBP	62.14	21207	26410	20.7264	161,238,948
BGLBP	89.93	<b>21156</b>	<b>26307</b>	<b>20.7365</b>	<b>161,208,067</b>

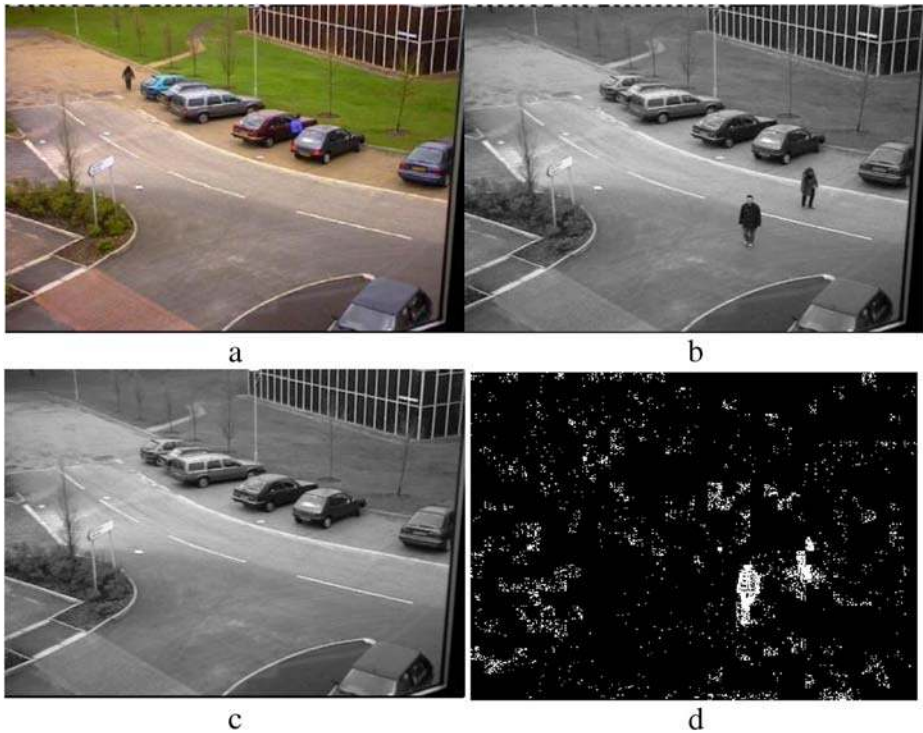
The values in bold are the best results in each column

**Table 6** Performance of each LBP version using the SM dataset without initialization

Method	Time (ms)	FDR (pixels)	$\chi^2$	PSNR	RMS (pixels)
Original LBP	72.04	22332	34292	20.295	167,469,610
MB Block	58.95	22303	34406	20.2886	167,486,171
CMB Block	<b>41.53</b>	22301	34556	20.2816	167,571,330
ULPB	53.29	22285	34465	20.2866	167,483,899
ICS-LBP	64.41	22309	34294	20.2962	167,419,353
ID-LBP	72.834	22306	34300	20.2946	167,426,883
RIU-LBP	73.15	22310	34299	20.2944	167,428,572
Circular LBP	62.36	22304	34333	20.2921	167,437,950
BGLBP	91.37	<b>22255</b>	<b>34251</b>	<b>20.2962</b>	<b>167,40n19</b>

The values in bold are the best results in each column

objects, objects leaving the scene, and the movements of moving objects affect the performance of different LBP methods. It is clear that the average value of the desired factors cannot highlight these effects in detail. In addition, the efficiency of the methods such as the evolutionary algorithms and the history-based methods do not appear in a short time. Since the utilized background extraction method is a history-based method, it is not logical to expect a significant improvement. Second, in this research, the LBP descriptor is not used to describe



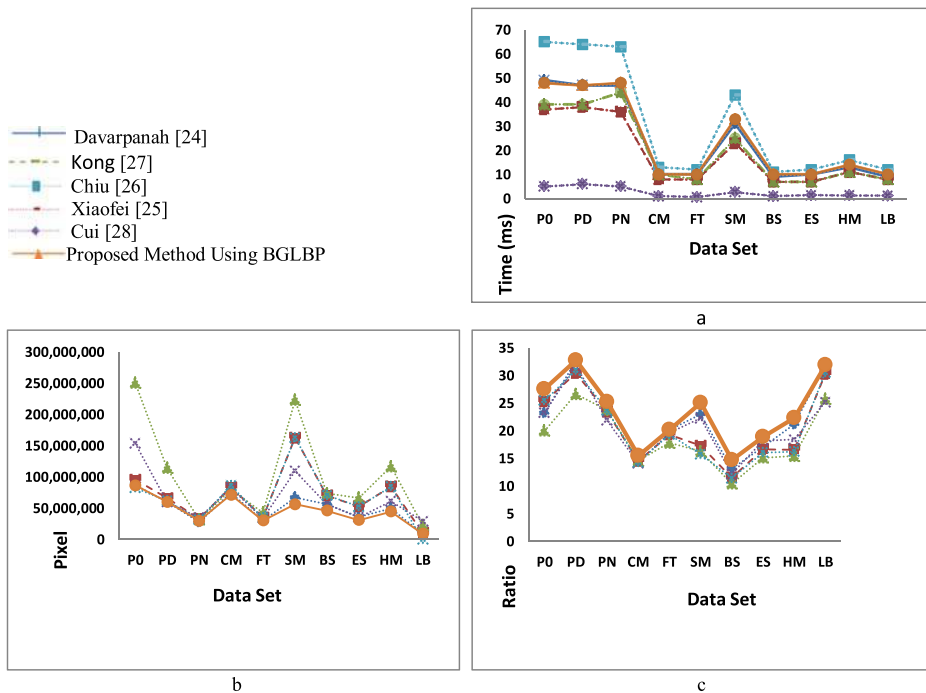
**Fig. 2** Detecting primary moving object by the proposed method using PETS2000 from 780th frame to 1280th. First frame (a), current frame (b), extracted final background (c), and last extracted binary image (d)

the details of the texture. LBP is only used to help the system to distinguish the background blocks from the foreground blocks. This means that the enhancement of the LBP version does not have a significant effect on the extracted background.

## 5.2 Analysis of BGLBP in the background extraction vs. other extraction methods

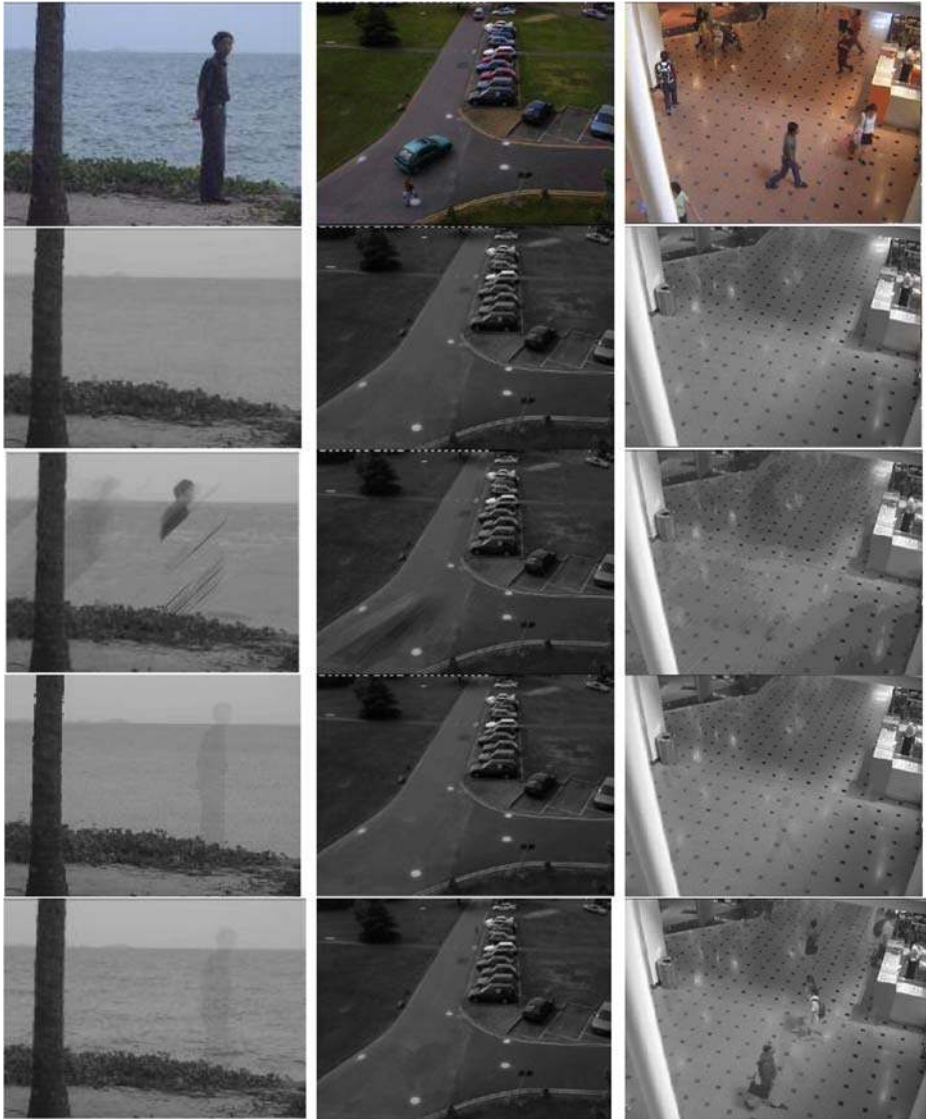
This experiment was done to investigate the performance of our background extraction method, which was presented in [6] using BGLBP. Overall, the following steps were applied in this experiment:

- The background was initialized using the proposed initialization method in [5].
- Each frame was divided into a number of non-overlapped blocks.
- BGLBP was calculated for each pixel.
- Each block of the current frame was classified as a foreground block or a background block according to the history of each block and using a history-based method. Historical information of each block was a set of the BGLBP histograms.
- If a block was detected as background, the model's history had to be updated. Otherwise, it would be determined whether each pixel in the block was a foreground pixel or a background pixel. For this purpose, a mixture of Gaussian model was used. According to the number of detected foreground pixels, the block was categorized again as a foreground block or a background block.
- An adapted method was used to update the history of pixels and blocks depending on whether they were detected as a background or a foreground.

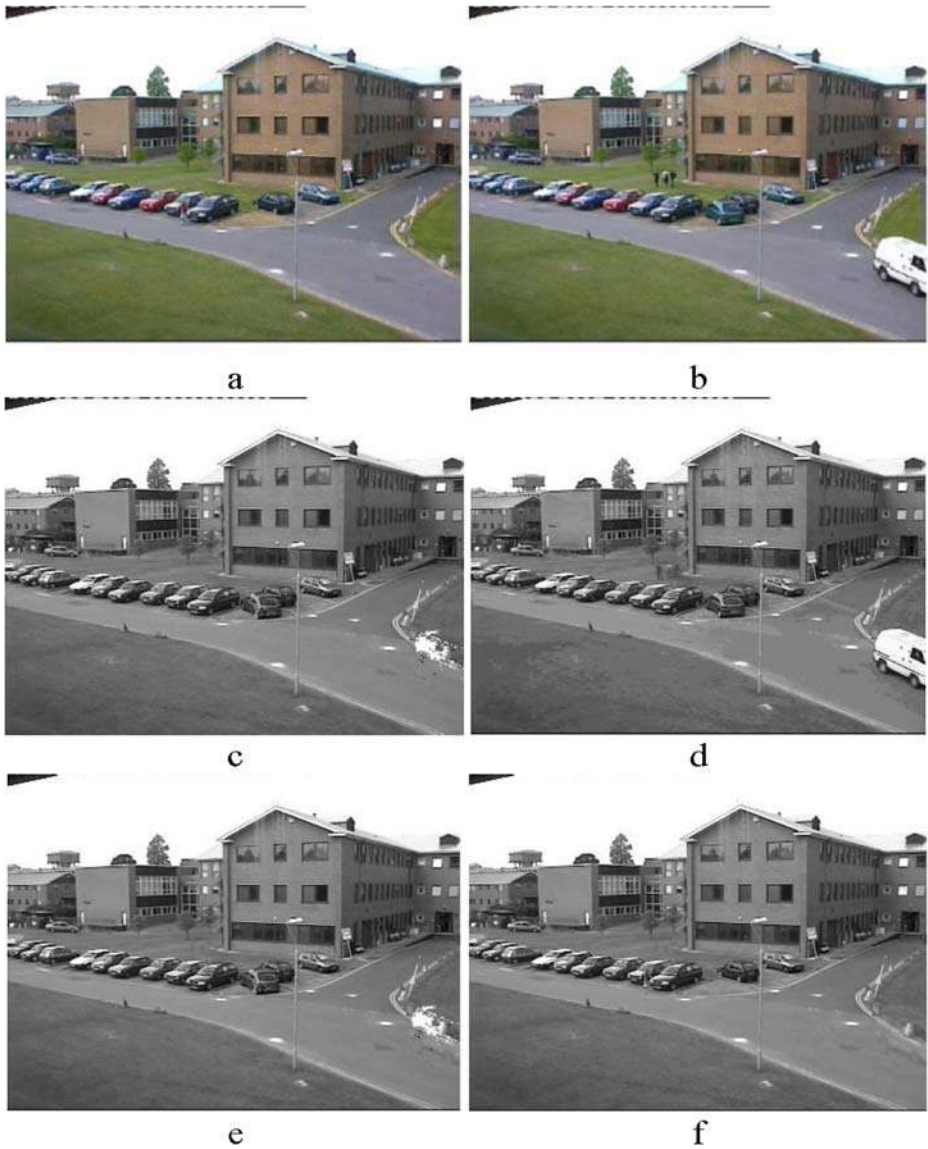


**Fig. 3** Comparison between proposed method with BGLBP and state-of-the-art **a** execution Time, **b** root mean square error, **c** peak signal to noise ratio of different methods

- Moving pixels were identified.
- A simple method was used to extract the final background image.
- $X_2$ , FDR, PSNR, and RMS form the desired factors of the experiment. These values were calculated from the extracted background image.
- The results were compared with three recent methods Xiaofei [23], Chiu [3], Davarpanah [6], Kong [13], and Cui [4].

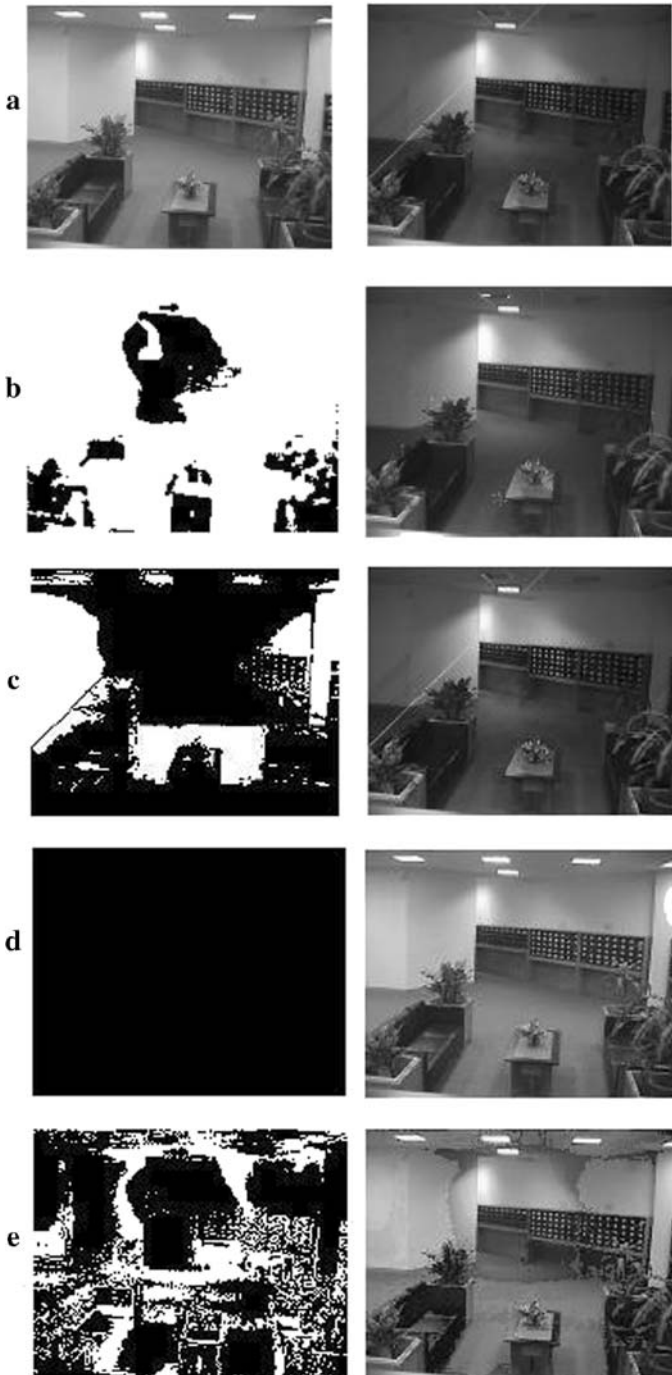


**Fig. 4** Quality of the extracted backgrounds by the methods. Cols: the water surface frame sequence, the PETS2001 night frame sequence, and the shopping mall frame sequence respectively. Rows: current frame, the extracted background by the proposed method, by Xiaofei [23], by Chiu [3], and by Kong [13] respectively



**Fig. 5** Detecting entered object being static by the methods using PETS2001 test-day dataset from 400th frame to 1300th. 400th frame (a), 1300th frame (b), extracted final background by proposed method (c), extracted final background by Xiaofei [23] (d), extracted final background by Chiu [3] (e), extracted final background by Kong [13] (f)

The datasets here were PETS2000 (P0), PETS2001 at day (PD), PETS2001 at night (PN), Campus (CM), Fountain (FT), Shopping Mall (SM), Bootstrap (BS), Escalator (ES), Hall Monitor (HM), and Lobby (LB). The average of the time consumption of these methods was shown in Fig. 2a. As it is clear from this figure, the time consumption of Chiu [3] is better in comparison to the other methods. Actually, the time complexity of their main method (the Extraction Background method) is  $O(N^2)$  versus the other methods with a time complexity equal to  $O(N^3)$ .  $N$  is size of each



**Fig. 6** Detecting a huge illumination change by the methods using lobby dataset from 200th frame to 1000th frame. **a** First and last frames. **b** Extracted binary image and final background image by the proposed method. **c** Extracted binary image and final background image by Kong [13]. **d** Extracted binary image and final background image by Chiu [3]. **e** Extracted binary image and final background image by Xiaofei [23].

dimension of the images. The average of RMS values in all the methods is illustrated in Fig. 2b. This figure shows that the error values in the presented method are less than those for the other methods. The mean value of PSNR values in six methods are plotted in Fig. 2c. Since PSNR explains the peak of signal to noise, the bigger values of this parameter are more valuable. The results show the presented method has an acceptable performance in comparison to other ones.

Figure 3 shows the extracted background using our method and the state-of-the-art using different frame sequences. It is completely clear that ours method has been more successful.

In addition, in order to evaluate qualitatively the resulted background extraction image by applying BGLBP, a number of tests were conducted. In each experiment, a specific kind of illumination change was raised and it was shown that the proposed method is able to overcome these problems.

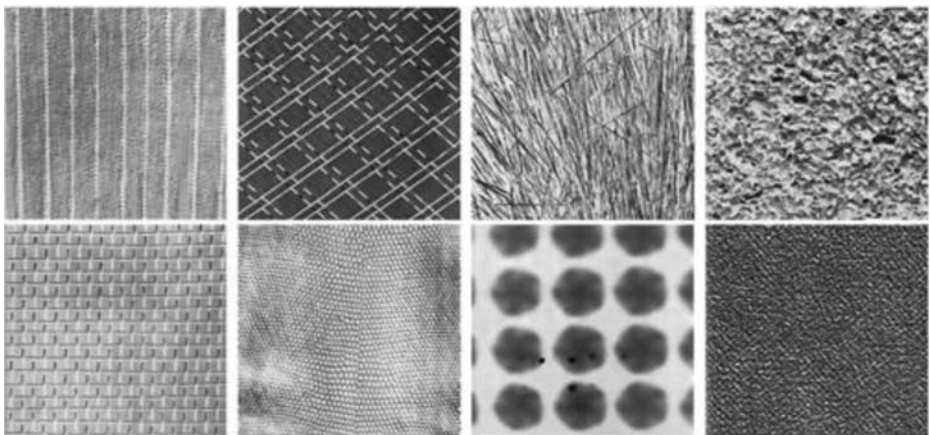
### 5.2.1 *There are moving objects in the first frame*

Sometimes there is a moving object in the first frame and when the history of background model is being initialized. It is expected that the method detects the moving object and removes it from the extracted background. In order to assess the performance of the proposed method in this state, a test using PETS 2000 (PO) dataset is done. As it is clear in Fig. 4, the proposed method has detected and removed the pedestrian in the top left corner of the first frame.

### 5.2.2 *Static objects are added to the scene*

Sometimes an entered moving object stops in the frame sequence. After being stable for a period, the method should detect this object as a part of the extracted background. Clearly, it is not possible to detect the object immediately; usually this section of the background image is constructed evolutionarily and pixel by pixel or block by block. In addition, methods adapt the intensity value of remaining object pixels gradually. The construction speed is dependent on the adaptability rate of the method.

In order to check the performance of the proposed method in these situations, four methods were executed using PETS 2001 test-day (PD) dataset. To achieve this situation, 400th frame to 1300th frame were utilized. It can be seen in Fig. 5 that Kong [13] could not detect the parked vehicle at the center of the picture as a stable object in the extracted background in the 1300th



**Fig. 7** A number of Brodatz images

frame. In addition, there is a newly parked van in the bottom right corner. Except Kong [13], the other methods constructed it as a statistic object in the final background. Compared to the others, a more section of van was constructed in the extracted background image by our method.

### 5.2.3 A big illumination change is accrued

Sometimes a big illumination change is accrued. It is expected that the change is detected fast and correctly, and the extracted background to get updated gradually. In facing with this condition, methods which are not enough relied on the history change their extracted background suddenly so that a sudden big noise is able to disturb their results completely. on the other side, methods whose results are depended completely to the history neglect the changes until a long time. Our method performs well in these conditions because of using BGLBP as its texture descriptor and using blocking to handle the background updating. Figure 6 illustrates this mater. Two frames in the first row shows a big illumination change between the first and the last frames (a lamp has turned off). It is clear in the images of the second row that our method using BGLBP has been able to detect changes correctly (whit pixels in the binary image) and to update background image gradually after about 300 frames.

## 5.3 Analysis of BGLBP as a texture descriptor

### 5.3.1 BGLBP compared to the other LBP versions

BGLBP deserves to be used in different applications. In this experiment, BGLBP was tested as a texture descriptor. Overall, an experiment with the following structure was conducted:

- 42 gray-level texture images of the Brodatz dataset ( $640 \times 640$ ) were randomly selected from [2] and used as dataset. Figure 7 presents a number of used images from this dataset.
- Each image was divided into  $4 \times 4$  non-overlapped sub-images. Except for the first sub-image of each image, which was used as the reference for that texture class, the other sub-images participate in the experiment separately. This means that the total number of participant images in the experiment was  $42 \times 15$ .
- It was assumed that all the  $4 \times 4$  non-overlapped sub-images of each frame had the same texture.

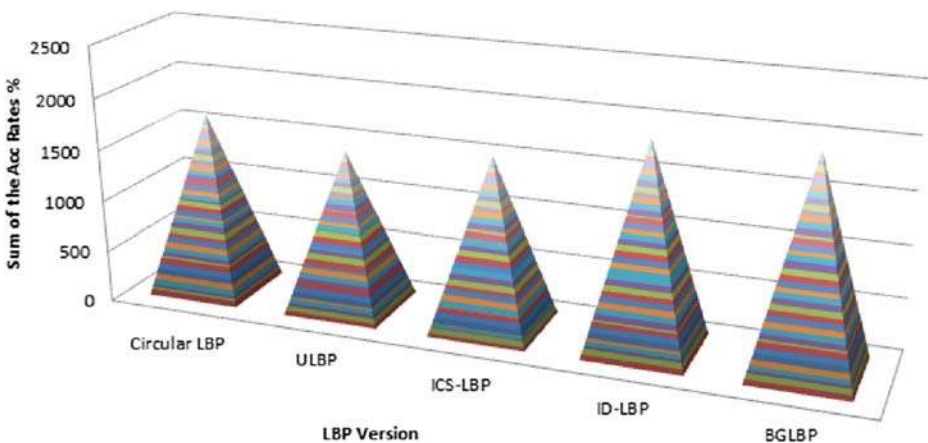


Fig. 8 Texture classification using different LBP versions



**Table 7** The Accuracy Rate of BGLBP (47.78 %) Compared to the others

	Circular LBP	ULBP	ICS-LBP	ID-LBP
Accuracy rate (%)	41.11	36.19	38.89	46.51
Improvement (%)	6.67	11.59	8.89	1.27

- The L1-distance was used as the measurement criterion to calculate the distance between each two histograms.
- A sub-image was classified to the nearest texture class with the minimum L1-distance between its histogram and histograms of the 42 first sub-images.

The accuracy of texture classification was calculated for Circular LBP, ULBP, ICS-LBP, ID-LBP, and BGLBP. Figure 8 plots the achieved results for each of the mentioned LBP versions. This figure shows that BGLBP is more stable rather than the other versions throughout the images with different textures and different homogeneous levels. In this figure, each cone belongs to a LBP version and each layer inside the cones, presented with different colors, refers to an image so that the vertical axis shows the sum of accuracies in terms of percentage.

Table 7 lists the resulted accuracy values for different versions of LBP. In this table, the average of various versions' accuracy values (the number of correctly classified samples over the total number) were calculated along 42 the participate images and they were compared with the corresponding average for BGLBP. The second row highlights the improvement from using BGLBP instead of the other versions of LBP. From these values, it can be concluded that the average of the accuracy values for BGLBP is 47.78 %, while this value for other methods is 40.67 %.

A paired *t*-test was performed to reveal the reliable difference between the accuracy rates of BGLBP and that of the other four versions. The comparison results with these methods for the probability value for a 95 % confidence level were shown in Table 8. In these test, all 42 images were participated.

### 5.3.2 BGLBP is less noise sensitive

As a texture descriptor, BGLBP is less noise sensitive. We carried out the following test to show BGLBP's less noise sensitivity compared the other versions of LBP. We conducted the test in the same manner as the previous test in Section 5.3.1. However, we applied a Salt and Pepper noise to the image with different noise density from 0 to 0.1. 50 reference images were used. The experiments were done on 4 selected images being divided each to 16 blocks. The only factor in selecting these 4 images was their diversity (see Fig. 9). LBP histograms were calculated for each block and it was classified as a reference image with the minimum histogram distance. Finally, the recall factor was calculated for all 16 blocks. The results were shown Fig. 10.

**Table 8** The paired *t*-test of BGLBP as a texture classifier compared to a number of LBP versions

	With circular LBP	With ULBP	With ICS-LBP	With ID-LBP
t-value	3.993	5.099	4.383	0.813
p-value	2.64005996546834E-4	8.1644477547086E-6	7.940544311313092E-5	0.4208021693727002

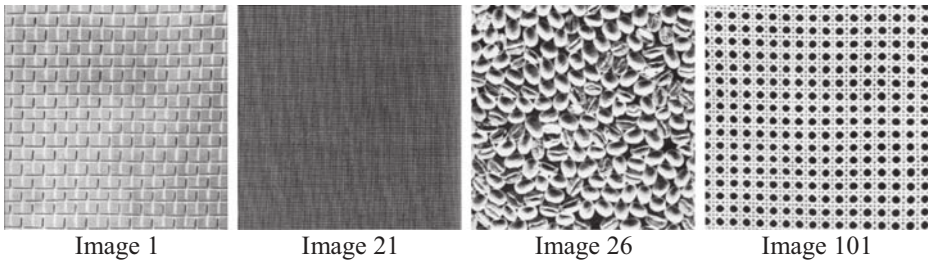


Fig. 9 Sample images used in the test

### 6 Conclusion

This paper introduced a new version of LBP (BGLBP), which was an extension of different LBP versions, and designed to handle a greater amount of illumination changes in the background extraction process. The evaluation of BGLBP was done in three different ways. It was evaluated as a block distinguisher in the background extraction process in comparison with the other versions of LBP, and as an independent texture classifier. It was also evaluated as a block distinguisher in the background extraction process in comparison with a number of benchmarks. The results showed that not only is the performance of the background extraction process improved using BGLBP but also BGLBP has the potential to be used as a texture descriptor in other applications.

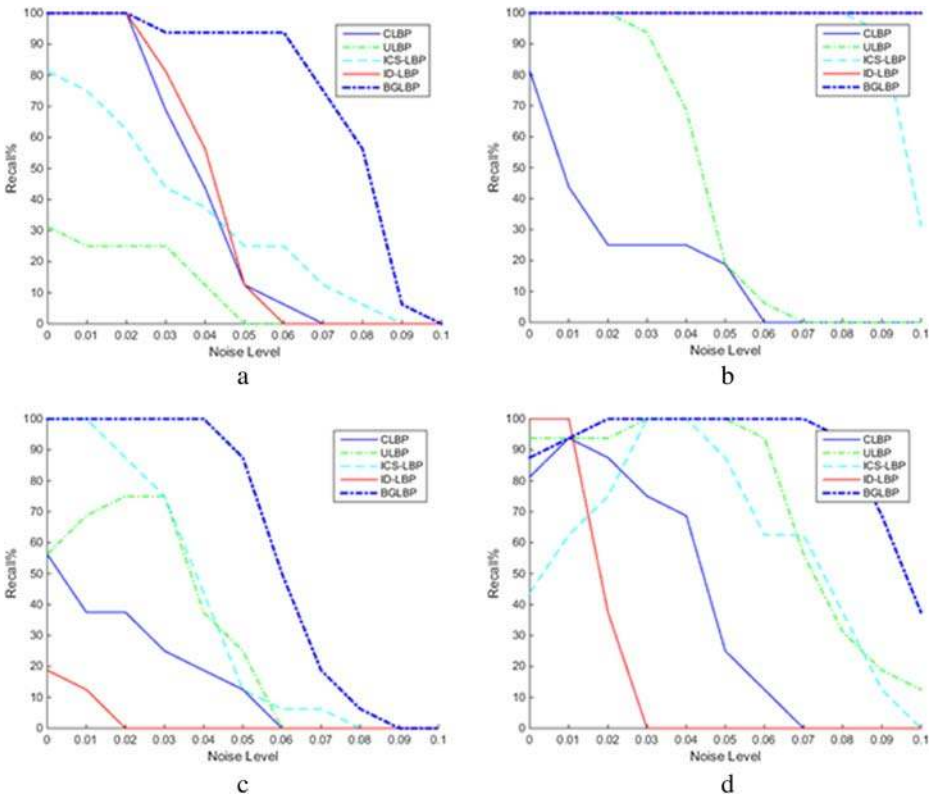


Fig. 10 Recall values via different noise densities in CLBP, ULBP, ICS-LBP, ID-LBP, and BGLBP

## References

1. Armanfard N, Komeili M, Valizadeh M, Kabir E, Jalili S (2009) A non-parametric pixel-based background modeling for dynamic scenes, in International Conference on Advances in Computational Tools for Engineering Applications (ACTEA '09) 2009, pp. 369–373
2. Brodatz textures, <http://www.uix.uis.no/~tranden/brodatz.html>, Ed., ed, 1999
3. Chiu C-C, Ku M-Y, Liang L-W (2010) A robust object segmentation system using a probability-based background extraction algorithm. *IEEE Trans Circ Syst Video Technol* 20:518–528
4. Cui B-x, Sun S-m, Duan Y (2009) Vehicle detection based on adaptive background, in Second International Workshop on Knowledge Discovery and Data Mining (WKDD), 2009, pp. 821–824
5. Davarpanah SH, Fatimah K, Lili NA, Puteri SS, Golchin M (2012) Using multi-scale filtering to initialize a background extraction model. *J Comput Sci* 8:1077–1084
6. Davarpanah SH, Fatimah K, Golchin M (2012) An effective history-based background extraction system. *J Comput Sci* 8:1062–1069
7. He X, Li J, Jia W, Wu Q, Hintz T (2007) Local binary patterns on hexagonal image structure, in 7th IEEE International Conference on Computer and Information Technology (CIT), 2007, pp. 639–644
8. He Y, Sang N (2011) Robust illumination invariant texture classification using gradient local binary patterns, in International Workshop on Multi-Platform/Multi-Sensor Remote Sensing and Mapping (M2RSM), 2011, pp. 1–6
9. Heikkilä M, Pietikainen M, Schmid C (2009) Description of interest regions with local binary patterns. *Pattern Recogn* 42:425–436
10. Heikkilä M, Pietikainen M (2006) A texture-based method for modeling the background and detecting moving objects. *IEEE Trans Pattern Anal Mach Intell* 28:657–662
11. I. f. I. Research, “A-star”, [http://perception.i2r.a-star.edu.sg/bk\\_model/bk\\_index.html](http://perception.i2r.a-star.edu.sg/bk_model/bk_index.html), Ed., ed, 2000
12. Junding S, Shisong Z, Xiaosheng W (2010) Image retrieval based on an improved CS-LBP descriptor, in The 2nd IEEE International Conference on Information Management and Engineering (ICIME), 2010, pp. 115–117
13. Kong J, Zheng Y, Lu Y, Zhang B (2007) A novel background extraction and updating algorithm for vehicle detection and tracking, in Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), 2007, pp. 464–468
14. Liao W-H (2010) Region description using extended local ternary patterns, in 20th International Conference on Pattern Recognition (ICPR), 2010, pp. 1003–1006
15. Liao W-H, Young T.-J (2010) Texture classification using uniform extended local ternary patterns, in IEEE International Symposium on Multimedia (ISM), 2010, pp. 191–195
16. Luo J, Fang Y, Cai Q (2010) A MANOVA of major factors of RIU-LBP feature for face recognition, in 20th International Conference on Pattern Recognition (ICPR), 2010, pp. 1028–1031
17. Mason M, Duric Z (2001) Using histograms to detect and track objects in color video, in 30th Applied Imagery Pattern Recognition Workshop, 2001, pp. 154–159
18. Ojala T, Pietikainen M, Harwood D (1994) Performance evaluation of texture measures with classification based on Kullback discrimination of distributions, in Conference A: Computer Vision & Image Processing., Proceedings of the 12th IAPR International Conference on Pattern Recognition., 1994, pp. 582–585
19. Ojala T, Pietikainen M, Maenpää T (2002) Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans Pattern Anal Mach Intell* 24:971–987
20. t. I. W. o. P. E. o. T. a. Surveillance (2009) PETS: performance evaluation of tracking and surveillance, <ftp://ftp.pets.rdg.ac.uk>, Ed., ed, 2009
21. Wu Q-Z, Cheng H-Y, Fan K-C (2004) Motion detection based on two-piece linear approximation for cumulative histograms of ratio images in intelligent transportation systems, in IEEE International Conference on Networking, Sensing and Control, 2004, pp. 309–314
22. Xia D, Sun H, Shen Z (2010) Real-time infrared pedestrian detection based on multi-block LBP, in International Conference on Computer Application and System Modeling (ICCASM), 2010, pp. V12-139-V12-142
23. Xiaofei L (2009) A new fast hybrid algorithm of background extraction suitable for the DSP Platform, in WRI World Congress on Computer Science and Information Engineering, 2009, pp. 1–4
24. Xiaosheng W, Junding S (2009) An effective texture spectrum descriptor, in Fifth International Conference on Information Assurance and Security (IAS '09) 2009, pp. 361–364
25. Xue G, Sun J, Song L (2010) Dynamic background subtraction based on spatial extended center-symmetric local binary pattern, in IEEE International Conference on Multimedia and Expo (ICME), 2010, pp. 1050–1054

26. Yang X, Yao X, Li D, Cai L (2010) Adaptive image retrieval based on generalized Gaussian model and LBP, in IEEE 2nd Symposium on Web Society (SWS), 2010, pp. 103–107
27. Zhang L, Chu R, Xiang S, Liao S, Li SZ (2007) Face detection based on multi-block LBP representation. In: Lee S-W, Li S (eds) Advances in biometrics. vol. 4642. Springer, Berlin, pp 11–18
28. Zhao S, Gao Y (2008) Establishing point correspondence using multidirectional binary pattern for face recognition, in 19th International Conference on Pattern Recognition (ICPR), 2008, pp. 1–4
29. Zhao G, Pietikainen M (2006) Local binary pattern descriptors for dynamic texture recognition, in 18th International Conference on Pattern Recognition (ICPR), 2006, pp. 211–214



**Seyed Hashem Davarpanah** received a B.E. degree in computer engineering in 2001 from Ferdowsi University, Iran, an M.S. degree in Artificial Intelligence and Robotics from IUST, Iran in 2003, and a Ph.D degree in Intelligent Computing from UPM, Malaysia in 2013. He is an Assistant Professor of Computer Engineering at USC, Iran. He has cooperated with this university from 2004. His research interestes include image processing and analysis, medical image processing, and evolutionary algorithms.



**Fatimah Khalid** obtained the B. Sc. In Computer Science from University Technology Malaysia (UTM) in 1992. During 1993 to 1995, she works as a System Analyst at University Kebangsaan Malaysia (UKM) and continued her Master's Degree from UKM. After getting her Masters Degree in 1977, she started involved in teaching at Sal College until 1999 and continued at the University Putra Malaysia in June 1999. She received her PhD in System Science and Management from the University Kebangsaan Malaysia in 2008. Currently she is working in Faculty of Computer Science and Information Technology as a lecturer with Associate Professor.



**Lili Nurliyana Abdullah** completed her PhD from Universiti Kebangsaan Malaysia in 2008. Currently she is Associate Professor at the department of Multimedia, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia. Her research interest includes video retrieval, computer games and animation.



**Maryam Golchin** received her B.E. degree in computer engineering in 2006 from Sajjad University, Iran, and a M.S. degree in Computer Vision from UPM, Malaysia in 2013. She is currently pursuing a Ph.D. degree in Computer Engineering from Griffith Univeristy, Australia. Her research interests include image processing and analysis, computer vision, and medical image processing.