Journal of Engineering and Applied Sciences 12 (21): 5478-5484, 2017
ISSN: 1816-949X

An Efficient Leaf (Texture) Classification using Local Binary Pattern with Noise Correction

[1]Anil Kumar Muthevi and [2]Ravi Babu Uppu
[1]Department of Computer Science and Engineering, Aditya College of Engineering and Technology, Acharya Nagarjuna University, Surampalem, Andhra Pradesh (AP), India
[2]Department of Computer Science and Engineering, Narasimha Reddy Engineering College, Secunderabad, Telangana (TS), India

Abstract: Leaf classification by using images based on their textures is the main objective of this study. Local Binary Pattern (LBP) operator is eminent extracting method but it is not effective especially in the cases where noise (noise occurs due to external sources and other reasons) in the images involved or corrupted the image patterns. Local Ternary Pattern (LTP) is another famous feature extracting method gives solution to some extent but not completely solves this problem. Towards achieving perfectness of classification by correcting noisy bits, we propose a method for both error detection and correction called Corrected LBP (CLBP) based on the analysis of uniform binary patterns which are appears more frequently in the natural images and almost all image structures. We suggested in our proposed method modification of bits in the pattern based on the analysis of neighbouring bits. It gives significant increase of accuracy and performance levels.

Key words: Texture, noise, local binary patterns, uniform patterns, local ternary patterns, LBP

INTRODUCTION

An image worth thousand words of information. Now a days by the help of digital image processing we can study and apply various methods to extract the hidden information from the images and analyse them to get useful results for various types of applications. In the recent years, after invention of modern digital and web cameras and cell phones there is rapid development in the aspects of speed and accuracy in the area of digital image processing.

The word "texture" plays critical role in the field of image processing and machine vision. Even though there is no exact definition for the term texture accepted it is gaining popularity from several decades. Classification is the process of segregating things in to various well known categories according to certain criteria. There are several methodologies developed on texture classification still there is a scope to extend or modify them. Still for many researchers it is one of the challenging and interesting topic.

The texture classification has wide variety of applications in diverged areas. Here, we are mentioning some of them. For stone classification we can recognize various types of stones and their quality especially where human inspection or observation is difficult or impossible (at underground, mines, etc.). We can apply texture classification methods not only for face recognition we can also classify them into various age groups or gender classification. By using textures in the field of remote sensing classify the places in certain area as hills, grounds, fields and rivers, etc. In medical image processing we can recognize the tissues of the skin based on their texture which are effected by skin deceases and to identify breast cancer by using image textures (these pictures are called mammograms). In timber industry by wood classification we can identify the strong and quality wood. In textile industry there is quality checking of fabrics based on the inspection of their properties like texture, colour and thickness. The application areas are not only limited to above but also content-based feature extraction from image and video databases, motion and activity analysis and biometrics.

Leaf is one of the important part of the plant or a tree for classification criteria because it always available irrespective of seasons and time. Whereas, some flowers and fruits of the trees are not available for a particular time or season. So, classification of plants and trees based on their leaves gives more effectiveness than other parts of the tree. Leaf classification is to identify and separate the leaves based on their appearance and structure and colour. Leaf classification in scientific methods is a

Corresponding Author: Ravi Babu Uppu, Department of Computer Science and Engineering, Narasimha Reddy Engineering College, Secunderabad, Telangana (TS), India

challenging and time consuming process for scientists and botanists. Leaf identification or classification has crucial role in grading of quality leafs in tea and tobacco industries. There is a much progressive work done by previous researchers in this area until now. Let us discuss the previous methods briefly. Aakif and Khan (2015) developed an automated method of leaf classification based on their leaves which uses morphological features like Aspect Ratio (AR) and Eccentricity (E) and Roundness (R) and convex hull features. Lakshmi and Mohan (2016), proposed a method to detect leaf types based on the shape and colour along with texture features. By using artificial intelligent methods which are applied on binary images of leavesresearched by Horaisova and Kukal (2016). By using template-based method Anjomshoae and Rahim (2016) identified overlapping rubber tree leaves. Recognition of the leaf image set based on the method of manifold-manifold distance done by Du *et al.* (2016).

To classify textures there are two major categories of algorithms called statistical and structural methods. The Local Binary Pattern (LBP) features which are invented by Pietikainen *et al.* (2011) is a simple but plays an crucial role in texture classification. It comes under category of structural method. At initial stage many reviewers treated it as ad hoc method with no theoretical foundation. Due to its simplicity in computational operations and its discrimination power it is very popular approach in many applications such as industrial inspections. Later, there are so many variations of LBP developed. Mean LBP which consider the effects of central pixels (Jin *et al.*, 2004, 2005; Ruiz-del-Solar and Quinteros, 2008; Hamood *et al.*, 2016). Hamming LBP (Yang and Wang, 2007) incorporate different (non-uniform) texture patterns into similar (uniform) patterns. Extended LBP which gives discriminate the same local binary patterns, cause high dimensionality rotation invariant LBP (Huang *et al.*, 2006, 2007). Completed LBP of textures for leaf classification by Muthevi and Uppu (2017) which gives importance to magnitude component also along with sign component (Guo *et al.*, 2010) for getting more accurate and efficient results. Soft LBP which is not invariant to monotonic grayscale changes and gives high computational complexity (Ahonen and Pietikainen, 2007). Multi-block LBP which capture micro and macro structure information (Zhang *et al.*, 2007; Liao *et al.*, 2007). It considers multiple block of LBP code. Elongated LBP which extract the anisotropic information and lose anisotropic information; not invariant to rotation (Liao and Chung, 2007). In Local Ternary Pattern (LTP) in the place of binary code, the pixel difference is encoded as 3 different valued code according to pre-defined threshold value. When compared to LBP it is less sensitive to noise. Today due to the growth of LBP is not an ordinary texture operator, it is treated as foundation for researchers dealing with local binary image and video description.

This study, is aimed to classify the leaves according to their LBP texture features with noise resistance for leaf classification.

MATERIALS AND METHODS

Origin of LBP: LBP uses only 4 neighbours called $LBP_{4,1}$ while $LBP_{16,2}$ takes 16 neighbours with radius 2. The original Local Binary Pattern code (LBP code) which encodes the local structure around each pixel. Here each picture element compared with its surrounding eight neighbours by subtracting the central pixel value. If the resulting value is strictly negative then encoded with 0 and otherwise fill with ones. A binary number formed by padding all these binary digits in clock wise direction starting from top of the left most corner and its equivalent decimal value used for labelling. Then obtained binary values are referred as Local Binary code or simply LBP code.

We can determine LBP operator for different sizes of neighbourhood sizes. Generally, the notation $LBP_{P,R}$ stands for P equally spaced adjacent pixels on a circle of radius R that forms symmetric neighbour set in the circular form. It produces the total no of 2^P different output values, corresponding to the 2^P different binary patterns respectively. These can be formed by P pixels in the neighbour set (Fig. 1), for example:

(a)

11	15	38
19	26	28
100	22	57

(b)

-15	-11	12
-7	-	2
74	4	-31

(c)

15	11	12
7	-	2
74	4	31

(d)

-1	-1	1
-1	-	1
1	1	-1

Fig. 1: a): A 3×3 sample blocks; b) local difference; c) magnitude component and d) sign component

There is small no of bitwise transitions from 0-1 and vice-versa of binary pattern (Pietikainen *et al*., 2011) defines them as fundamental patterns or uniform patterns. For example, 11111111 and 00000000 contains no transitions (null transitions) while 00001100 and 00111100 contains 2 transitions and so on. While, 01010101 contains more than 2 transitions called non-uniform transitions. In the natural images we can observe uniform patterns are formed more frequently than non-uniform patterns. The main problem with non-uniform patterns is they harms the classification performance.

We can observe that in uniform LBP noise is less forming LBP histogram. But LBP is sensitive to the noise. A small noise will causes different encoded pixels. The main advantages of LBP are:

- It is simple to implement but efficient and widely used in many applications
- It has well texture discrimination characteristic
- LBP is very strong to monotonic gray scale changes

Demerits:

- During the process of image rotation, long histogram produced by LBP operator does not support
- Due to small differences in the pixels which are used to compute LBP and its variants are vulnerable to noise
- The LBP does not support high spatial support
- In practice, there is no guarantee for the independence of local differences between central pixel and its neighboring pixels
- In LBP only sign component plays important role as classification descriptor. And it will gives more information about the image

Local Ternary Patterns (LTP): Local Ternary Pattern (LTP) encoding the small pixel difference into a third state (Tan and Triggs, 2010). Local ternary patterns not completely solves the problem by introducing the third state in addition to the two states.

Let $C^T_{P,R} = b^T_{p-1}\ b^T_{p-2},\ldots, b^1_T b^T_0$ denotes the local ternary pattern code of P neighbors at the distance of R to the central pixel and $LTP_{P,R}$ denotes such scheme for $C^T_{P,R}$. Here, each bit can be obtained as:

$$
\begin{aligned}
b^T_p &= 1,\ \text{if } z_p = t,\\
&= 0, \text{if } |z_p| < t,\\
&= -1, \text{if } Z_p \leq -t
\end{aligned}
$$

Here, the value called as t is a pre-defined threshold.

78	99	53
54	54	60
67	54	60

The ternary code for the above values are (by taking t = 2 as threshold value):

1	1	0
0	-	1
1	-1	-1

The corresponding positive LBP component is (11010010):

1	1	0
0	-	1
1	0	0

The corresponding negative LBP component is (11110011):

0	0	0
0	-	0
0	1	1

LTP gives more resistant to noise but it has very large dimensionality of LTP histogram, so by Tan and Triggs (2010). LTP is decomposed as positive LBP component and a negative LBP component. The positive LBP component bits can be obtained as:

$$
\begin{aligned}
b^P_p &= 1, \text{if } z_p \geq t\\
&= 0, \text{if } z_p > t
\end{aligned}
$$

The negative LBP component bits can be obtained as:

$$
\begin{aligned}
b^P_p &= 1, \text{if } z_p \leq -t,\\
&= 0, \text{if } z_p > -t
\end{aligned}
$$

LTP is the composition of positive and negative Local Binary Pattern. When compared to LBP, LTP has doubles the no of patterns.

Noise causes the small pixel differences. There is no mechanism to correct the corrected patterns. To solve this problem, there is a Noise-Resistant LBP (NRLBP) and Extended Noise-Resistant LBP (ENRLBP) proposed by Ren *et al*. (2013) is major inspiration to our present research, we apply to classify on leaves.

Corrected LBP (CLBP): In proposed Corrected Local Binary Pattern code (CLBP), encode the small pixel differences as uncertain bit 'X' then find out the value of 'X' decided on the other certain bits of LBP:

$$b_p^N = 1, \text{ if } Z_p \geq t,$$
$$= X, |Z_p| \leq t,$$
$$= 0, Z_p \geq -t$$

State 0, 1 represents two definite (strong) states. Where the pixel difference is almost definitely negative and positive, respectively. Noise can unlikely change from 0-1 or vice versa. We can coded with an uncertain state regardless its sign.

The value of the uncertain bit can be consider into either 0 or 1, represented by a variable x_i, x_i belongs to {0, 1}. Let, X = {x_1, x_2, x_3, ..., x_n} denotes the vector format by n variable of a code, x belongs to $\{0, 1\}^n$. The uncertain code can be denoted by n variables of a code:

$$b_{p-1}^N\ b_{p-2}^N,\ldots,\ b_1^N\ b_0^N = C(X)$$

After detecting the uncertain code, we calculate the uncertain bits based on the values of the other certain bits to from one or more codes of image local structure.

Since uniform (similar) patterns occurs more likely than non-uniform ones, we assign the value of uncertain bit x so as to form possible uniform LBP code. For example, we can find out the uncertain bit of uncertain code "11x1x0x0" so, the possible uniform patterns are, 1111100 and 11110000.

Let 'm' denote the no of elements in S_{CLBP} if m>0, the bin corresponding to each element in S_{CLBP} will be added by 1/m. All these pattern generate from one uncertain code. If m = 0, the non-uniform bin will be added by 1 this is applicable for each and every pixel in the patch.

Case 1; for m = 1:

78	99	53
30	54	60
54	12	13

LBP code; 11010010:

- The LTP codes are: 11010000, 11110010
- The uncertain code becomes: 11X100X0
- By using correction code (CLBP) we can get: 11110000

Case 2; for m = 2:

78	99	53
30	54	60
54	12	54

LBP code; 11011010:

- The LTP codes are: 11010000, 11111010
- The uncertain code becomes: 11X1X0X0
- By using correction code (CLBP) we can get: 11110000, 11111000

Case 3; for m = 3:

78	54	99
30	54	54
54	12	53

LBP code; 11110010:

- The LTP codes are: 10100010, 11111010
- The uncertain code becomes: 1X1XX0X0
- By using correction code (CLBP) we can get: 11100000, 11110000 and 11111000

Case 4; for m = 4:

78	99	60
53	54	54
67	53	13

LBP code; 11110010:

- The LTP codes are: 11100010, 11110111
- The uncertain code becomes: 111X0X1X
- By using correction code (CLBP) we can get: 11100011, 11100111, 11110011, 11110111

Case 5; for m = 0:

78	12	53
54	54	60
67	12	54

LBP code; 10011011:

- The LTP codes are: 10010010, 10111011
- The uncertain code becomes: 10X1X01X
- By using correction code (CLBP) we can get: NO uniform code possible

In all above cases, the proposed CLBP corrects back to uniform code, when the noise change the uniform pattern code to unstable non-uniform pattern (Fig. 2). The proposed CLBP different from LBP and LTP with the following scenario:

- It is more capable of noise-resistant
- The disturbed LBP code can be corrected back to uniform code
- For LTP uncertain bits are set to 1 for negative half and 0 for positive half but in corrected LBP they could code as 1/0 determined depends on other bits of the code
- LTP histogram consists of doubles the bins when compared to corrected LBP histogram (Algorithm 1)

Algorithm 1: The proposed process can be explained as:

Step 1: Gathering the required leaf by using digital camera or obtained from standard leaf database and choose suitable images

Step 2: In this phase do the necessary preprocessing operations like conversion form color to gray scale images and choose the required part if necessary (cropping of the image)

For Each and every pixel

Step 3: Derive the local binary pattern code and local ternary pattern by using threshold value

Step 4: Derive the uncertainty bits in the code C(X)

Search uncertain bits X in the space $\{0,1\}^n$ so that, c(X) forms uniform LBP codes

Step 5: Based on the case we can correct the code by using the

Step 6: Construct the histogram

If m value is equal to zero accumulate the non-uniform bin with 1

Else Accumulate the bin of each pattern in S_{NRLBP} with 1/m

Step 7: Apply classification method Nearest-Neighbor (NN) classifier with 3 different distance measures called Chi-square, histogram insertion and G-Statistic

Step 8: Compare the results with already existing methods

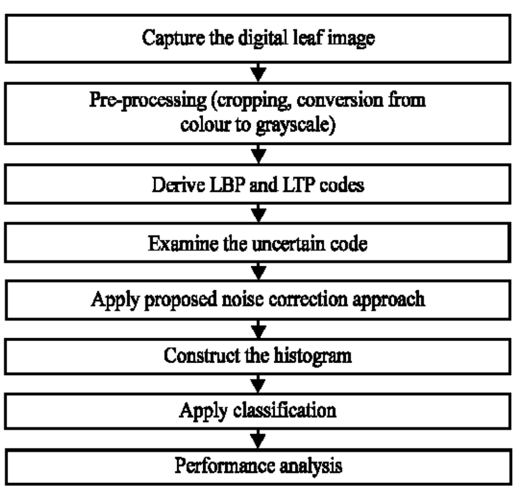

Fig. 2: Block diagram (over view) of the proposed method

RESULTS AND DISCUSSION

Experimental setup: The experiments conducted by us to test the performance of the proposed correction method. The proposed approach also compared with uniform LBP and uniform LTP on leaf recognition. In our experiments, for leaf recognition, we use the Nearest-Neighbour (NN) classifier with different distance measures. For leaf classification we choose a challenging experimental setup. We have taken a sequence of experiments with the following datasets. Some of them available to public.

AFF dataset: This database contains forty two varieties of leaves with different shapes and colors. The experiments consists of 134 leaf images on the pure white back ground. Here are some images along with their scientific names (Fig. 3).

Foliage dataset: This database contains forty two varieties of leaves with different shapes and colors. The experiments consists of 25 leaves per plant for the training set and another twenty for the testing purpose. Here are some images along with their scientific names (Fig. 4).

Flavia leaf dataset: This database contains 1097 varieties of leaves with different shapes and colors from 32 plants. But in our experiments we have taken only 10 and 30 randomly selected images per class for test and training classes, respectively. Here are some images along with their scientific names (Fig. 5 and Table 1-3).

Table 1: The leaf recognition rate and optimal threshold on the Austrian Federal Forest (AFF) datasets

Name of the algorithm	Chi-square distance method (%)	Histogram intersection method (%)	Modified G-Statistics method (%)
LBP	95.07	92.32	95.12
LTP	98.25	96.98	98.29
DLBP	95.12	96.83	97.45
FLBP	98.45	96.16	97.54
Proposed corrected LBP	98.71	96.23	98.72

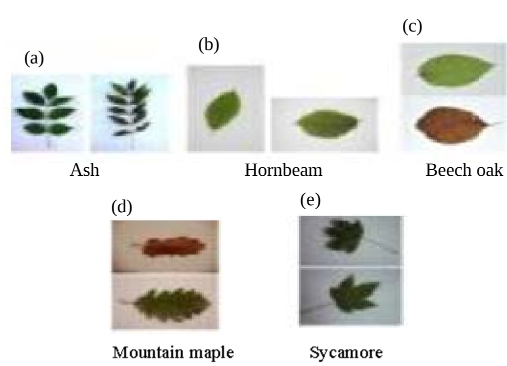

Fig. 3: Sample images of AFF dataset

Table 2: The leaf recognition rate and optimal threshold on the foliage dataset

Name of the algorithm	Chi-square distance (%)	Histogram intersection (%)	Modified G-Statistics (%)
LBP	75.07	72.32	75.12
LTP	78.25	76.98	78.29
DLBP	75.12	76.83	77.45
FLBP	78.45	76.16	77.54
Proposed corrected LBP	78.71	76.23	78.72

Table 3: The leaf recognition rate and optimal threshold on flavia leaf dataset

Name of the algorithm	Chi-square distance (%)	Histogram intersection (%)	Modified G-Statistics (%)
LBP	75.07	72.32	75.12
LTP	88.25	86.98	88.29
DLBP	85.12	86.83	87.45
FLBP	78.45	86.16	87.54
Proposed corrected LBP	88.71	86.23	88.72

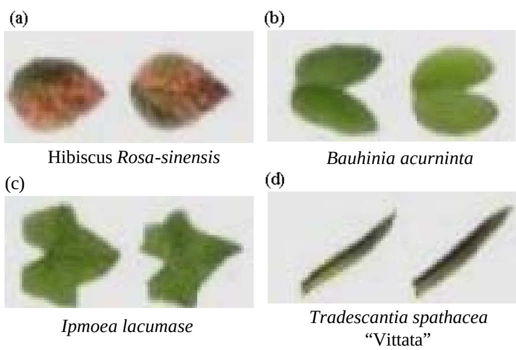

Fig. 4: Sample images of foliage dataset

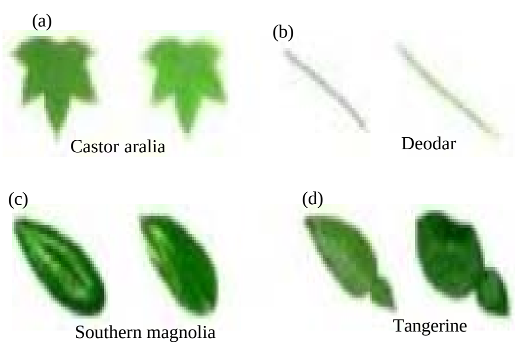

Fig. 5: Sample images of flavia dataset

CONCLUSION

Local binary pattern is very sensitive to the noise. Due to external noise LBP changes significantly. Local ternary pattern partially gives the solution. But in both cases (LBP and LTP) treat the corrupted patterns as it is, i.e., there is no recovery process for image local structures. In the proposed corrected (CLBP) method noisy non-uniform patterns are adjusted back to uniform code.

The proposed approaches show stronger when compared to other existing methods with respect to the noise resistance. It show the superior performance on the above leaf classification/recognition applications.

ACKNOWLEDGEMENT

The researchers would like to say thanks to their family members, management, principal, teaching and non-teaching staff and students for their continuous support and encouragement.

REFERENCES

Aakif, A. and M.F. Khan, 2015. Automatic classification of plants based on their leaves. Biosyst. Eng., 139: 66-75.

Ahonen, T. and M. Pietikainen, 2007. Soft histograms for local binary patterns. Proceedings of the Symposium on Finnish Signal Processing (FINSIG'07) Vol. 5, August 30, 2007, University of Oulu, Oulu, Finland, pp: 1-4.

Anjomshoae, S.T. and M.S.M. Rahim, 2016. Enhancement of template-based method for overlapping rubber tree leaf identification. Comput. Electron. Agric., 122: 176-184.

Du, J.X., M.W. Shao, C.M. Zhai, J. Wang and Y. Tang *et al.*, 2016. Recognition of leaf image set based on manifold-manifold distance. Neurocomputing, 188: 131-138.

Guo, Z., L. Zhang and D. Zhang, 2010. A completed modeling of local binary pattern operator for texture classification. IEEE Trans. Image Process., 19: 1657-1663.

Hamood, S.F., M.S.M. Rahim, O. Farook and D. Kasmuni, 2016. A survey on various deblurring methods. J. Eng. Appl. Sci., 11: 561-569.

Horaisova, K. and J. Kukal, 2016. Leaf classification from binary image via artificial intelligence. Biosyst. Eng., 142: 83-100.

Huang, D., Y. Wang and Y. Wang, 2007. A robust method for near infrared face recognition based on extended local binary pattern. Proceedings of the 3rd International Symposium on Advances in Visual Computing (ISVC 2007), November 26-28, 2007, Springer, Lake Tahoe, ?Nevada, USA., pp: 437-446.

Huang, Y., Y. Wang and T. Tan, 2006. Combining statistics of geometrical and correlative features for 3D face recognition. Proceedings of the Conference on British Machine Vision, September 4-7, 2006, Imperial College London, London, England, UK., pp: 879-888.

Jin, H., Q. Liu, H. Lu and X. Tong, 2004. Face detection using improved LBP under Bayesian framework. Proceedings of the 3rd International Conference on Image and Graphics (ICIG'04), December 18-20, 2004, IEEE, Hong Kong, China, ISBN:0-7695-2244-0, pp: 306-309.

Jin, H., Q. Liu, X. Tang and H. Lu, 2005. Learning local descriptors for face detection. Proceedings of the IEEE International Conference on Multimedia and Expo (ICME05), July 6-6, 2005, IEEE, Amsterdam, Netherlands, ISBN:0-7803-9331-7, pp: 928-931.

Lakshmi, B.V. and V. Mohan, 2016. Kernel-based PSO and FRVM: An automatic plant leaf type detection using texture, shape and color features. Comput. Electron. Agric., 125: 99-112.

Liao, S. and A.C. Chung, 2007. Face recognition by using elongated local binary patterns with average maximum distance gradient magnitude. Proceedings of the 8th Asian Conference on Computer Vision, November 18-22, 2007, Springer, Tokyo, Japan, pp: 672-679.

Liao, S., X. Zhu, Z. Lei, L. Zhang and S.Z. Li, 2007. Learning multi-scale block local binary patterns for face recognition. Proceedings of the International Conference on Biometrics (ICB 2007), August 27-29, 2007, Springer, Seoul, Korea, pp: 828-837.

Muthevi, A. and R.B. Uppu, 2017. Leaf classification using completed local binary pattern of textures. Proceedings of the IEEE 7th International Conference on Advance Computing (IACC), January 5-7, 2017, IEEE, Hyderabad, India, ISBN:978-1-5090-1561-0, pp: 870-874.

Pietikainen, M., A. Hadid, G. Zhao and T. Ahonen, 2011. Computer Vision Using Local Binary Patterns. Vol. 40, Computational Imaging and Vision, Springer-Verlag, London, ISBN: 0857297473, Pages: 211.

Ren, J., X. Jiang and J. Yuan, 2013. Noise-resistant local binary pattern with an embedded error-correction mechanism. IEEE. Trans. Image Process., 22: 4049-4060.

Ruiz-del-Solar, J. and J. Quinteros, 2008. Illumination compensation and normalization in eigenspace-based face recognition: A comparative study of different pre-processing approaches. Pattern Recognit. Lett., 29: 1966-1979.

Tan, X. and B. Triggs, 2010. Enhanced local texture feature sets for face recognition under difficult lighting conditions. IEEE. Trans. Image Process., 19: 1635-1650.

Yang, H. and Y. Wang, 2007. A LBP-based face recognition method with Hamming distance constraint. Proceedings of the 4th International Conference on Image and Graphics (ICIG07), August 22-24, 2007, IEEE, Sichuan, China, ISBN:978-0-7695-2929-5, pp: 645-649.

Zhang, L., R. Chu, S. Xiang, S. Liao and S. Li, 2007. Face detection based on Multi-Block LBP representation. Adv. Biometrics, 4642: 11-18.