Compound Local Binary Pattern (CLBP) for Rotation Invariant Texture Classification

Faisal Ahmed
Department of CIT, Islamic University of Technology, Gazipur, Bangladesh

Emam Hossain
Department of CSE, Ahsanullah University of Science and Technology, Dhaka, Bangladesh

A.S.M. Hossain Bari
Samsung Bangladesh R & D Center Ltd, Dhaka, Bangladesh

Md. Sakhawat Hossen
Department of CIT, Islamic University of Technology, Gazipur, Bangladesh

ABSTRACT
The local binary pattern (LBP) provides a simple and efficient approach to gray-scale and rotation invariant texture classification. However, the LBP operator thresholds P neighbors at the value of the center pixel in a local neighborhood and employs a P-bit binary pattern to encode only the signs of the differences between the gray values. Thus, the LBP operator discards some important texture information. In this paper, we have proposed the compound local binary pattern (CLBP), an extension of the LBP texture operator for rotation invariant texture classification. The CLBP operator exploits $2P$ bits to encode the information of a local neighborhood of P neighbors, where the extra P bits are used to express the magnitude information of the differences between the center and the neighbor gray values. A feature representation method based on CLBP codes is presented. Experimental results show that, the classification rate of the proposed method is appreciable.

General Terms
Image Processing, Pattern Recognition.

Keywords
Compound local binary pattern, local binary pattern, support vector machine, texture classification, Brodatz album.

1. INTRODUCTION
Texture classification is an active research topic that has been widely studied due to its potential applicability in fabric inspection, remote sensing, and medical image analysis [1]. However, classification in uncontrolled environment is a major challenge as textures in the real world tend to be non-uniform due to variations in illumination, orientation, scale, or other visual appearances [2]. Early methods for texture classification were mostly based on statistical analysis of the texture images to characterize the stochastic properties of the spatial distributions of gray levels [3]. Some of the common methods include the gray tone co-occurrence matrix and filtering based approaches which provide good classification results with sample texture images of identical or similar orientations [1]. However, in practice performance of statistical methods decreases significantly in presence of orientation variations in textures [1], suggesting the need for rotation invariant texture classification methods.

The first few approaches for achieving rotation invariance in texture description include generalized co-occurrence matrices [4], anisotropy features [5] and circular autoregressive model [6] based methods. Later, texture classification based on hidden Markov model [7], Gaussian Markov random field [8] and auto correlation methods [9] were introduced that provide invariance to texture orientations. Many feature based methods that exploit Gabor wavelets or other basis functions were also introduced for rotation invariant texture classification [10], [11], [12], where rotation invariance was attained by either computing rotation invariant features or converting rotation variant features into rotation invariant ones [2]. Recently, Ojala et al. [2] has proposed a gray-scale and rotation invariant texture classification method based on local binary pattern (LBP). The LBP method extracts rotation invariant texture features from a local region by thresholding the gray values of the P-neighborhood pixels relative to the corresponding value of the central pixel, which is computationally efficient and robust to monotonic illumination variation. Although LBP provides a theoretically simple, yet efficient approach to texture classification, it has some limitations. Firstly, it shows poor performance in the presence of random noise [13]. Secondly, LBP method only considers the sign of the difference between two gray values and thus discards the magnitude of the difference which is very important texture information. Hafiane et al. [14] has proposed median binary pattern (MBP) that provides robustness against noise as texture primitives are obtained by thresholding a 3×3 neighborhood against the local median. On the other hand, local ternary pattern (LTP) introduced by Tan and Triggs [15] employs an extra discrimination level than LBP in order to provide robustness in smooth regions. This method has been successfully applied in face recognition. A similar method has been proposed by He and Cercone [16] for content-based image retrieval. Zhou et al. [13] has proposed an extended LBP that classifies and combines the non-uniform local patterns based on their structure and occurrence probability in order to compensate the texture information discarded by the original LBP operator. More recently, Guo et al. [1] has proposed LBP variance (LBPV) to characterize the local contrast information into the one-dimensional LBP histogram.

In this paper, we have proposed the compound local binary pattern (CLBP), an extension of the LBP operator for rotation invariant texture classification. Unlike the original LBP operator that uses P bits to encode only the signs of the differences between the center pixel and P neighbor gray values, the proposed method employs $2P$ bits, where the additional P bits are used to encode the magnitude information of the differences between the center and the neighbor gray values in a local neighborhood using a threshold. The motivation behind the proposed encoding scheme is to increase the robustness of the texture feature representation by incorporating additional information that is discarded by the original LBP operator. We empirically study the effectiveness of our proposed method in representing texture information. The performance of the CLBP feature representation is evaluated in terms of classification rate using support vector machine. Experiments with a widely-used texture image database, namely the Brodatz texture album [17], demonstrate that, the proposed CLBP operator is more robust in extracting texture information and provides higher classification rate compared to some existing feature representation techniques.

2. LOCAL BINARY PATTERN (LBP)

LBP is a gray-scale and rotation invariant texture primitive that describes the spatial structure of the local texture of an image. The LBP operator selects a local neighborhood around each pixel of an image, thresholds the P neighbor gray values with respect to the center pixel and concatenates the result binomially. The resulting binary value is then assigned to the center pixel. Formally, LBP operator takes the following form:

$$LBP_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} s(i_p - i_c)2^p \quad (1)$$

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

Here, i_c is the gray value of the center pixel (x_c, y_c), i_p is the gray value of its neighbors, P is the number of neighbors and R is the radius of the neighborhood. The basic LBP encoding process is illustrated in Figure 1.

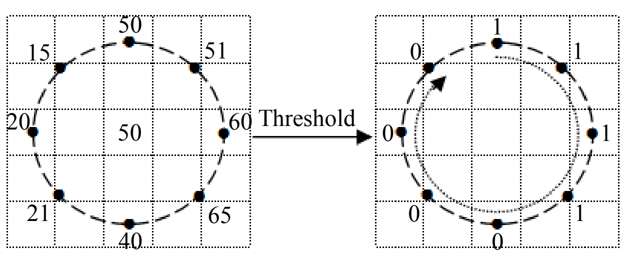

Figure 1: Illustration of the basic LBP operator. Here, the LBP code is 11110000.

To remove the effect of rotation, each binary pattern generated by the LBP operator is converted into a rotation invariant pattern using (3).

$$LBP_{P,R}^{ri} = min\{ROR(LBP_{P,R}, i) \mid i = 0,1,2,...,P-1\} \quad (3)$$

Here, $ROR(x, i)$ performs a circular bitwise right shift on a P-bit binary number x for i times. In practice, the LBP operator considers the signs of the differences of the gray values of P equally spaced neighbors with respect to the central pixel, which is then represented using a P-bit binary number. If any neighbor does not fall exactly on a pixel position, then the value of that neighbor is estimated using bilinear interpolation. The histogram of the encoded image block obtained by applying the LBP operator is then used as a texture descriptor for that block.

One extension to the original LBP operator, known as the uniform LBP (ULBP), exploits certain LBP patterns, which appear more frequently in a significant area of the image. These patterns are known as the uniform patterns as they contain very few bitwise transitions from 0 to 1 or vice versa in a circular sequence of bits. One example of a uniform pattern is 00011111. It has only one transition from 0 to 1. Ojala et al. [2] observed that, uniform LBP patterns are the fundamental properties of texture, which provide a vast majority of all the 3×3 patterns present in any texture image. Therefore, uniform patterns are able to describe significant local texture information, such as bright spot, flat area or dark spot, and edges of varying positive and negative curvature [2].

3. COMPOUND LOCAL BINARY PATTERN (CLBP)

The original LBP operator discards the magnitude information of the difference between the center and the neighbor gray values in a local neighborhood. As a result, this method tends to produce inconsistent codes. One example is shown in Figure 2. Here, the 8-bit uniform LBP code (11111111) corresponds to a flat area or a dark spot at the center pixel [16], which is not correct in this case.

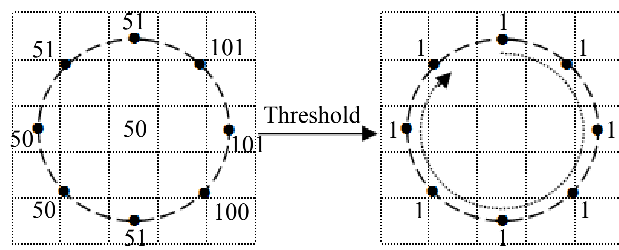

Figure 2: Generation of inconsistent binary pattern in LBP encoding process.

As LBP operator considers only the sign of the difference between two gray values, it often fails to generate appropriate binary code. Being motivated by this, we propose CLBP, an extension of the original LBP operator that assigns a $2P$-bit code to the center pixel based on the gray values of a local neighborhood comprising P neighbors. Unlike the LBP that employs one bit for each neighbor to express only the sign of the difference between the center and the corresponding neighbor gray values, the proposed method uses two bits for each neighbor in order to represent the sign as well as the magnitude information of the difference between the center and the neighbor gray values. Here, the first bit represents the sign of the difference between the center and the corresponding neighbor gray values like the basic LBP pattern and the other bit is used to encode the magnitude of the difference with respect to a threshold value, the average magnitude (M_{avg}) of the difference between the center and the neighbor gray values in the local neighborhood of interest. The CLBP operator sets this bit to 1 if the magnitude of the difference between the center and the corresponding neighbor is greater than the threshold M_{avg}.

Otherwise, it is set to 0. Thus, the indicator $s(x)$ of (2) is replaced by the following function:

$$s(i_p, i_c) = \begin{cases} 00 & i_p - i_c < 0, \quad |i_p - i_c| \le M_{avg} \\ 01 & i_p - i_c < 0, \quad |i_p - i_c| > M_{avg} \\ 10 & i_p - i_c \ge 0, \quad |i_p - i_c| \le M_{avg} \\ 11 & \text{otherwise} \end{cases} \qquad (4)$$

Here, i_c is the gray value of the center pixel, i_p is the gray value of a neighbor p, and M_{avg} is the average magnitude of the difference between i_p and i_c in the local neighborhood. The CLBP operator is illustrated in Figure 3.

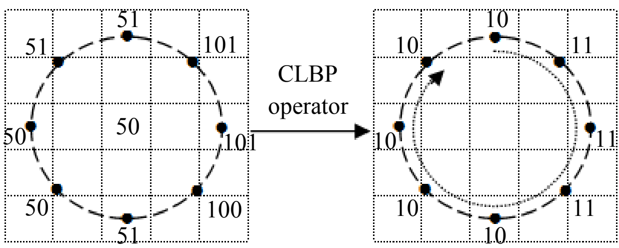

Figure 3: Illustration of the basic CLBP operator. Here, the generated CLBP code is 1011111110101010.

It can be observed that, the proposed method discriminates the neighbors in the north-east, east, and south-east directions as they have higher gray values than the other neighbors and thus produces a consistent local pattern.

4. SUB-CLBP CODE GENERATION

In a 3×3 neighborhood, the proposed CLBP method encodes an image by operating on the 8 neighbors around the central pixel and assigning a 16-bit code to that pixel. As 16-bit codes are used to label the pixels, the number of possible binary patterns is 2^{16}. To reduce the number of features, He and Cercone [16] have proposed to consider less number of neighbors while forming the binary patterns. Thus, this method discards some neighborhood information in order to reduce the length of the feature vector. Here, we have proposed a different approach where all the CLBP binary patterns are further split into two sub-CLBP patterns. Each sub-CLBP pattern is obtained by concatenating the bit values corresponding to $P/2$ neighbors, where P is the number of neighbors. Formally, in a local neighborhood, the two sub-CLBP patterns are formed by concatenating the corresponding values of the bit sequence (1, 2, 5, 6, …, $2P-3$, $2P-2$) and (3, 4, 7, 8, …, $2P-1$, $2P$), respectively of the $2P$-bit original CLBP code.

In other words, a 16-bit CLBP pattern is split into two 8-bit sub-CLBP patterns, where the first one is obtained by concatenating the bit values corresponding to the neighbors in the north, east, south, and west directions, respectively and the second sub-CLBP pattern is obtained by concatenating the bit values corresponding to the neighbors in the north-east, south-east, south-west, and north-west directions, respectively. Thus, this method reduces the number of possible patterns significantly, which results in a total of 2^8 distinct sub-CLBP patterns. The process is illustrated in Figure 4. The two sub-CLBP patterns are treated as separate binary codes and combined during the feature vector generation.

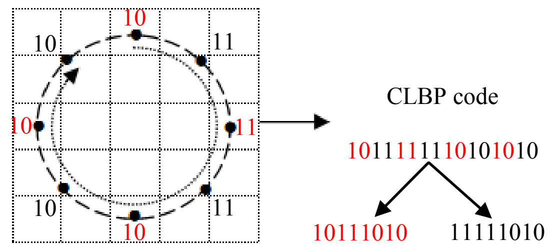

Figure 4: Generation of the sub-CLBP patterns from the original CLBP code.

5. FEATURE REPRESENTATION

After applying the CLBP operator on all the pixels of an image and splitting all the 16-bit CLBP patterns into the corresponding sub-CLBP patterns, we get two 8-bit binary codes for each pixel. In order to remove the effect of rotation, circular bitwise right shift is performed on each sub-pattern for $(P-1)$ times where P is the number of bits in the pattern and the minimum binary valued code is then selected from the results. Thus, all the sub-CLBP binary codes are converted into rotation invariant patterns using (5).

$$sCLBP^{ri} = min\{ROR(sCLBP, i) \mid i = 0, 1, 2, ..., P-1\} \qquad (5)$$

Here, $sCLBP$ is the P-bit sub-CLBP pattern obtained by splitting the original CLBP code, $sCLBP^{ri}$ is the rotation invariant code, and $ROR(x, i)$ performs a circular bitwise right shift on x for i times. Thus, two encoded image representations are obtained for the two sub-CLBP patterns. Two separate histograms are then calculated on the two encoded images, where each of the histograms comprises 2^8 bins. In order to obtain the feature vector, the two histograms are concatenated to obtain a spatially combined histogram. This combined histogram is used as the feature vector that represents the texture information of the image. Each feature vector contains 2×2^8 features, where some of the feature values may be 0. The overall process is shown in Figure 5.

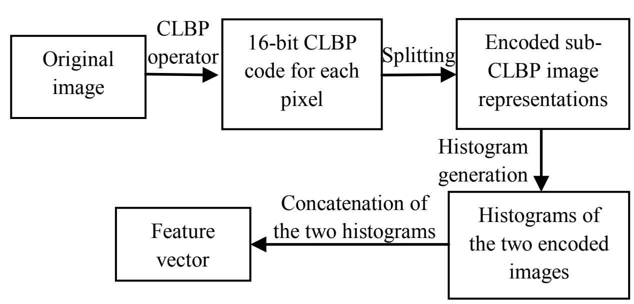

Figure 5: Proposed feature vector generation process using the CLBP operator.

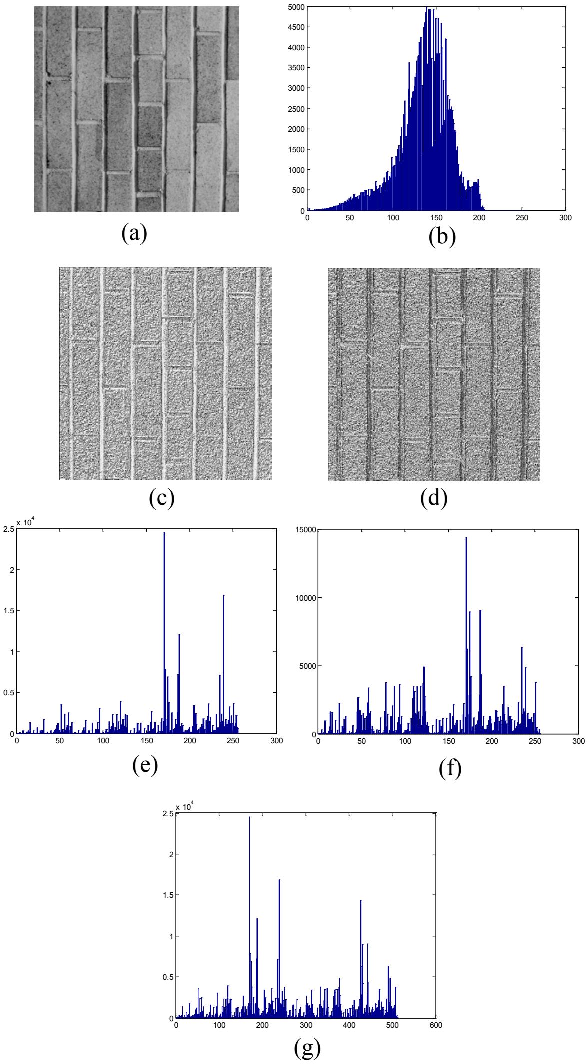

Figure 6: (a) is a sample texture image, (b) is the histogram of image (a), (c) and (d) are the encoded sub-CLBP images of (a), (e) and (f) are the histograms of image (c) and (d), (g) is the spatially combined histogram obtained by concatenating (e) and (f).

6. CLASSIFICATION USING SUPPORT VECTOR MACHINE (SVM)

Support vector machine (SVM) is a state-of-the-art machine learning approach based on modern statistical learning theory. It has been successfully applied in different classification problems. SVM performs classification by constructing a hyper plane in such a way that the separating margin between positive and negative examples is optimal. This separating hyper plane then works as the decision surface.

Given a set of labeled training samples $T = \{(\boldsymbol{x}_i, l_i), i = 1, 2, \ldots, L\}$, where $\boldsymbol{x}_i \in R^P$ and $l_i \in \{-1, 1\}$, a new test data $\boldsymbol{x}$ is classified by

$$f(\boldsymbol{x}) = \text{sign}(\sum_{i=1}^{L} \alpha_i l_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b) \tag{6}$$

Here, α_i are Lagrange multipliers of dual optimization problem, b is a threshold parameter, and K is a kernel function. The hyper plane maximizes the separating margin with respect to the training samples with $\alpha_i > 0$, which are called the support vectors. SVM makes binary decisions. To achieve multi-class classification, using one-against-rest classification or several two-class problems are the commonly used approaches. In our study, we used the one-against-rest approach. Radial basis function (RBF) kernel was used for the classification problem. The function K can be defined as

$$K(\boldsymbol{x}_i, \boldsymbol{x}) = \exp(-\gamma \, \| \boldsymbol{x}_i - \boldsymbol{x} \|^2), \quad \gamma > 0 \tag{7}$$

$$\| \boldsymbol{x}_i - \boldsymbol{x} \|^2 = (\boldsymbol{x}_i - \boldsymbol{x})^t (\boldsymbol{x}_i - \boldsymbol{x}) \tag{8}$$

Here, γ is a kernel parameter. A grid-search was carried out for selecting appropriate parameter value, as suggested in [18].

7. EXPERIMENTAL RESULTS

The proposed CLBP-based feature description method has been evaluated against a benchmark texture image database, namely the Brodatz album [17]. The experimental dataset included a total of 1050 gray-scale images with resolution of 256×256, 8 bits/pixel. The images are from 15 different texture classes, namely bark, brick, bubbles, grass, leather, pigskin, raffia, sand, straw, water, weave, wood, wool, canvas, and reptile. The source images were rotated to obtain 7 different rotation angles of 0°, 30°, 60°, 90°, 120°, 150°, and 200°. The dataset included 10 images for each rotation angle of a texture class.

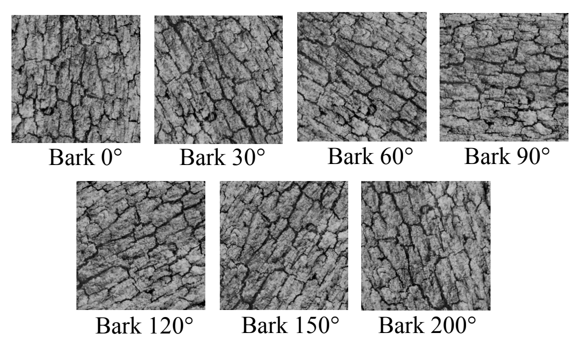

Figure 7: Sample images of a texture class digitized at different rotation angles.

In our experiment, half of the images in each class were used to train the classifier and the remaining images were used as the testing sets. Therefore, both the training and the testing dataset included 525 texture images of different rotation angles. We have compared the proposed method in terms of classification rate with some widely-used local texture operators, namely local binary pattern (LBP) [2], median binary pattern (MBP) [14], and local ternary pattern (LTP) [15]. Support vector machine was used for the classification task. Results obtained from the experiments are shown in Table 1. In all cases, the generated binary codes were converted to rotation invariant patterns using the method discussed in Section 5, which is indicated by the superscript *ri*.

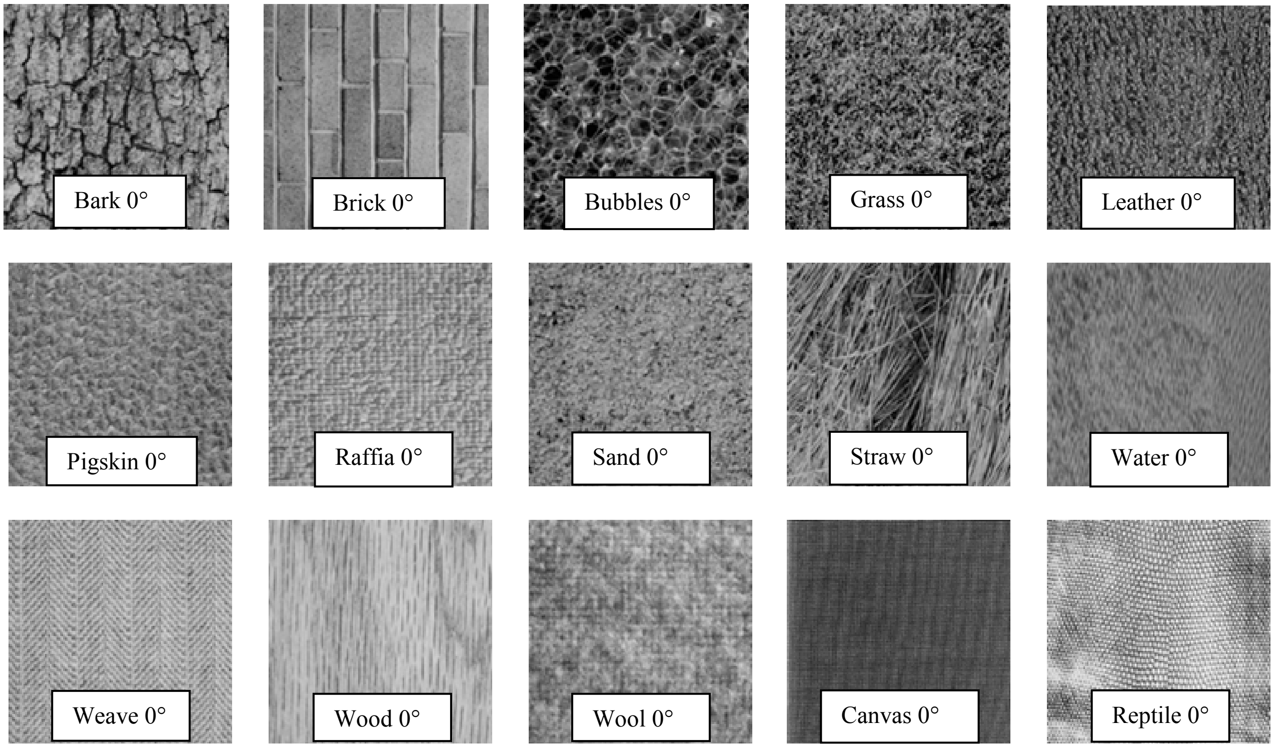

Figure 8: Sample images of 15 texture classes used for experiment.

Table 1. Classification rate of different texture operator using the Brodatz dataset

Local texture operator	No. of correctly classified images	Classification rate (%)
MBP^{ri}	415	79.04
$LBP^{ri}_{8,1}$	430	81.90
$LBP^{ri}_{16,2}$	438	83.43
$ULBP^{ri}_{8,1}$	448	85.33
LTP^{ri}	459	87.43
$CLBP^{ri}_{8,1}$	480	91.42
$CLBP^{ri}_{16,2}$	478	91.05

The classification rate of LBP and CLBP feature description methods can be influenced by adjusting two parameters: the number of selected neighbors P and the radius R of the local neighborhood. Therefore, we have evaluated the performance of LBP and CLBP method for different parameter values in order to find the optimal parameter setting. It can be observed that, CLBP provides the highest classification rate of 91.42% for the parameter setting $(P, R) = (8, 1)$. The highest classification rate obtained for different texture operators using the optimal parameter settings is shown in Figure 9.

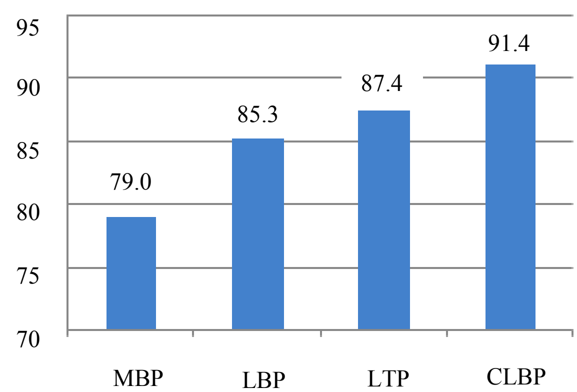

Figure 9: The highest classification rate (%) achieved for different local texture operators using optimal parameter setting.

From the experimental results, it is evident that, texture feature representation based on compound local binary pattern is more robust and this method provides higher classification rate than some existing methods for texture feature representation. The superiority of the CLBP encoding is due to the utilization of the magnitude of the difference between the center and the neighbor gray values by combining it with the basic LBP pattern, which acts as a compensation for the texture information discarded by the LBP operator. Thus, this method provides an effective and efficient approach to rotation invariant texture classification that is more discriminative than the original LBP operator and outperforms several existing texture description methods.

8. CONCLUSION

This paper describes the CLBP, an extension of the original LBP operator and a feature representation method based on CLBP codes for rotation invariant texture classification. The proposed method utilizes an encoding scheme that combines the magnitude information of the difference between two gray values with the original LBP pattern and thus provides increased robustness in many situations where LBP fails to generate consistent codes. Experimental results show that, the CLBP operator provides an effective and efficient approach for representing texture information with high discriminative ability, which is computationally efficient and robust against rotation effects.

9. REFERENCES

[1] Z. Guo, L. Zhang and D. Zhang, "Rotation invariant texture classification using LBP variance (LBPV) with global matching," Pattern Recognition, vol. 43, pp.706−719, 2010.

[2] T. Ojala, M. Pietikainen and T. Maenpaa, "Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns," IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, pp.971−987, 2002.

[3] U.S.N. Raju, A.S. Kumar, B. Mahesh and B.E. Reddy, "Texture Classification with High Order Local Pattern Descriptor: Local Derivative Pattern," Global Journal of Computer Science and Technology, vol. 10, issue 8, pp. 72−76, 2010.

[4] L.S. Davis, S.A. Johns and J.K. Aggarwal, "Texture Analysis Using Generalized Cooccurrence Matrices," IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 1, pp.251−259, 1979.

[5] D. Chetverikov, "Experiments in the Rotation-Invariant Texture Discrimination Using Anisotropy Features," Proceedings of the Sixth International Conference on Pattern Recognition, pp.1071−1073, 1982.

[6] R.L. Kashyap and A. Khotanzed, "A model-based method for rotation invariant texture classification," IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 8, no. 4, pp.472–481, 1986.

[7] W.R. Wu and S.C. Wei, "Rotation and gray-scale transform-invariant texture classification using spiral resampling, subband decomposition, and hidden Markov model," IEEE Transactions on Image Processing, vol. 5, no. 10, pp.1423–1434, 1996.

[8] H. Deng and D.A. Clausi, "Gaussian MRF rotation-invariant features for image classification," IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 26, no. 7, pp.951–955, 2004.

[9] P. Campisi, A. Neri, C. Panci and G. Scarano, "Robust rotation-invariant texture classification using a model based approach," IEEE Transaction on Image Processing, vol. 13, no. 6, pp.782–791, 2004.

[10] V. Manian and R. Vasquez, "Scaled and Rotated Texture Classification Using a Class of Basis Functions," Pattern Recognition, vol. 31, pp.1937−1948, 1998.

[11] N. Kim and S. Udpa, "Texture classification using rotated wavelet filters," IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, vol. 30, no. 6, pp.847–852, 2000.

[12] M. Kokare, P.K. Biswas and B.N. Chatterji, "Rotation-invariant Texture Image Retrieval using Rotated Complex Wavelet Filters," IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics, vol. 36, no.6, pp.1273–1282, 2006.

[13] H. Zhou, R. Wang and C. Wang, "A Novel Extended Local Binary Pattern Operator for Texture Analysis," Information Sciences, vol. 178, no. 22, pp.4314−4325, 2008.

[14] A. Hafiane, G. Seetharaman, and B. Zavidovique, "Median binary pattern for textures classification," Image Analysis and Recognition, pp. 387−398, 2007.

[15] X. Tan and B. Triggs, "Enhanced Local Texture Feature Sets for Face Recognition under Difficult Lighting Conditions," IEEE International Workshop on Analysis and Modeling of Faces and Gestures, LNCS 4778, pp.168-182, 2007.

[16] D. He and N. Cercone, "Local Triplet Pattern for content-based image retrieval," Image Analysis and Recognition, pp. 229–238, 2009.

[17] P. Brodatz, "Textures: A Photographic Album for Artists and Designers," Dover Publications, New York, 1966.

[18] C.W. Hsu and C.J. Lin, "A Comparison on Methods for Multiclass Support Vector Machines," IEEE Transaction on Neural Networks, vol. 13, no. 2, pp.415-425, 2002.