

Edge Information for Boosting Discriminating Power of Texture Retrieval Techniques

Abdelhamid Abdesselam

Department of Computer Science, Sultan Qaboos University, Oman
Email: ahamid@squ.edu.com

Abstract—Texture is a powerful image property for object and scene characterization, consequently, a large number of techniques has been developed for describing, classifying and retrieving texture images. On the other hand, edge information is proven to be an important cue used by the human visual system. Several physiological experiments have shown that, when looking at an object, human eyes explore different locations of that object through saccadic eye movements but they spend more time fixating edge regions. Based on this result, we hypothesize that a better performance could be obtained when analyzing an image (texture images in this case) if the visual features extracted from edge regions are given higher weights than those extracted from uniform regions. To check the validity of this hypothesis, we have modified several existing texture retrieval techniques in a way that incorporates the proposed idea and compared their performance with that of the original techniques. The results of the experiments that have been conducted on three common datasets confirmed the effectiveness of the proposed approach, since a significant improvement in the retrieval rate is obtained for all tested techniques. The experiments have also shown an improvement in the robustness to noise.

Index Terms—Texture Retrieval, Local Binary Patterns, Local Ternary Patterns, Local Binary Patterns Variance, 2D FFT, Discrete Wavelet Transform, Edge information.

I. INTRODUCTION

Texture is a powerful image property for object and scene characterization; it is easily perceived by human and provides a rich semantic information for many computer vision tasks such as image segmentation, scene classification, object recognition, and image retrieval; consequently texture analysis methods have been utilized in a variety of application domains such as automated defect detection and quality control, medical image analysis, and remote sensing, to name few [6].

On the other hand, edge information is an important cue used by various visual tasks. Shapley and Tolhurst [1] noted the importance of sharp contours in comprehending different aspects of perception. Marr [2] proposed a

computational theory on human vision in which edges constitute the most informative components in the image. Morrone and Burr [3] argued that edges and lines are particularly rich sources of image information, and that organization of these features could provide the basis for an efficient description of the image. Moreover, Tatler et al. [4] showed that “salient” regions such as those characterized by their contrast or orientation are more often fixated by observers; Baddeley and Tatler [5] investigated the possible factors guiding eye fixation; they found that fixation probability was dominated by high spatial frequency edge information;

A large number of techniques has been developed for describing, classifying and retrieving texture images. Many of these techniques have recognized the importance of edge information and considered it as a fundamental texture primitive. For example, Shapiro and Stockman [7] described a texture characterization method that combines a gradient histogram with edge orientation histogram. Zhang et al. [8] devised a technique, called Local Derivative Pattern (LDP) that uses higher order derivatives and an adequate binary coding to characterize textures. Local Binary Patterns technique and its variants [10-16] encode grayscale differences and build histograms out of those codes to describe the texture. Several multi-resolution techniques such as Gabor [19-20] and Discrete Wavelet [21-26] transforms, produce feature vectors that are mostly based on first and second order moments calculated on (sub-) images representing the original image contents at different frequencies and orientations. Similarly, several Fourier-based techniques [27-29] partition the frequency space into ranges (for example Low, medium and high frequency) from which feature vectors are generated.

In this paper, we describe an approach that makes use of edge information in a different way. We suggest introducing a preprocessing step before the actual feature extraction. In this step, edge pixels/regions are identified; Features extracted from edge pixels/regions are given higher weights than those extracted from non-edge pixels/regions. The proposed approach is mainly inspired from the research results reported in [5] suggesting that the human visual system pays more attention to regions rich in edges. To demonstrate the effectiveness of the proposed approach, we tested it on five image retrieval techniques.

The results showed a significant improvement in the retrieval performance of the modified techniques. Preliminary results on LBP variants have been already published in [31]. In this work, two additional techniques (FFT, and DWT) are considered and one more dataset (Vistex) is included in the experiments. Besides, the results of an experiment conducted for evaluating the robustness of the proposed approach against noise are included.

II. TECHNIQUES INCLUDED IN THIS INVESTIGATION

We have considered in this research work two types of techniques: global methods (represented by Fourier-based (FFT) and Wavelet-based (DWT) methods) and local methods (represented by three local binary patterns variants, LBP, LTP, and LBPV).

Fourier transform has been widely used for texture description. It has the very useful characteristic of highlighting the dominant spatial frequencies as well as the dominant orientations of the texture. Another advantage of Fourier transform is the fact that frequency-domain features are generally less sensitive to noise than the features in the spatial domain [27]. Several research works [27-29] partition the spectral domain into ring-shaped and wedge-shaped regions. The feature vector describing the texture is then made of means and variances of the Fourier power spectrum of those regions.

Formally, the discrete Fourier transform $F(u,v)$ of a $N \times N$ grayscale image $g(x,y)$ is defined by (1).

$$F(u, v) = \frac{1}{N^2} \sum_{x=0}^{x=n-1} \sum_{y=0}^{y=n-1} g(x, y) \cdot e^{-j2\pi(ux+vy)/N} \quad (1)$$

A ring region ($R_{r1,r2}$) between radius $r1$ and radius $r2$ is defined by (2).

$$R_{r1,r2} = \{(u,v) | r_1^2 \leq u^2 + v^2 < r_2^2; 0 \leq u \leq n-1; 0 \leq v \leq n-1\} \quad (2)$$

A wedge-shaped region between angles $\theta1$ and $\theta2$ is defined by (3).

$$W_{\theta1, \theta2} = \{(u,v) | \theta1 \leq \tan^{-1}(v/u) < \theta2, 0 \leq u \leq n-1; 0 \leq v \leq n-1\} \quad (3)$$

For a frequency value $F(u,v)$, which is in general complex, i.e. $F(u,v) = R(u,v) + j \cdot I(u,v)$, the power spectrum $P(u,v)$ is given by (4).

$$P(u, v) = |F(u, v)|^2 = R^2(u, v) + I^2(u, v) \quad (4)$$

The second global technique that has been considered in this work is the Discrete Wavelet Transform. There are

several physiological studies on the visual cortex that suggest the use of a multi-scale analysis of the visual information by the human visual system [30]. Besides, wavelet transform has the very useful property of space-frequency localization. Pyramidal wavelet decomposition is commonly used to define sub-bands at various resolutions. In this method, the low-frequency sub-band is repeatedly decomposed until a required level is reached. Means and standard deviations of the energy of the sub-bands are used as texture features. Formally, a discrete wavelet transform requires two functions; a scaling function usually separable $\phi(x, y) = \phi(x) \cdot \phi(y)$ and three directional wavelets (5).

$$\begin{aligned} \phi^H(x, y) &= \phi(x) \cdot \phi(y); \quad \phi^V(x, y) = \phi(x) \cdot \phi(y); \\ \phi^D(x, y) &= \phi(x) \cdot \phi(y) \end{aligned} \quad (5)$$

The DWT of a function $f(x,y)$ at level j is defined by (6) and (7).

$$W_\phi(j, m, n) = \frac{1}{\sqrt{N} \cdot N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cdot \phi_{j,m,n}(x, y) \quad (6)$$

(Low frequency component)

$$W^i_\phi(j, m, n) = \frac{1}{\sqrt{N} \cdot N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cdot \phi^i_{j,m,n}(x, y) \quad (7)$$

(High frequency components)

Where

$$i = \{H, V, D\}$$

$$\phi_{j,m,n}(x, y) = 2^{j/2} \phi(2^j x - m, 2^j y - n)$$

$$\phi^i_{j,m,n}(x, y) = 2^{j/2} \phi^i(2^j x - m, 2^j y - n)$$

Similarly, the first and second order statistics (means and standard deviations) are extracted from the low and high frequency components.

The third technique considered in this work is the Local Binary Patterns (LBP) approach which was first introduced by Ojala et al. in 1996 [10]. The LBP operator assigns to each image pixel g_c the decimal value of the binary string describing the local pattern around the pixel. The LBP code is formally defined by (8).

$$LBP(p, r) = \sum_{k=0}^{p-1} S(g_k - g_c) * 2^k \quad (8)$$

Where

$$S(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases}$$

g_k is the k^{th} neighbor of a central pixel g_c
 p is the number of neighbors considered
 r is the radius, specifying the distance between a central pixel and its neighbors

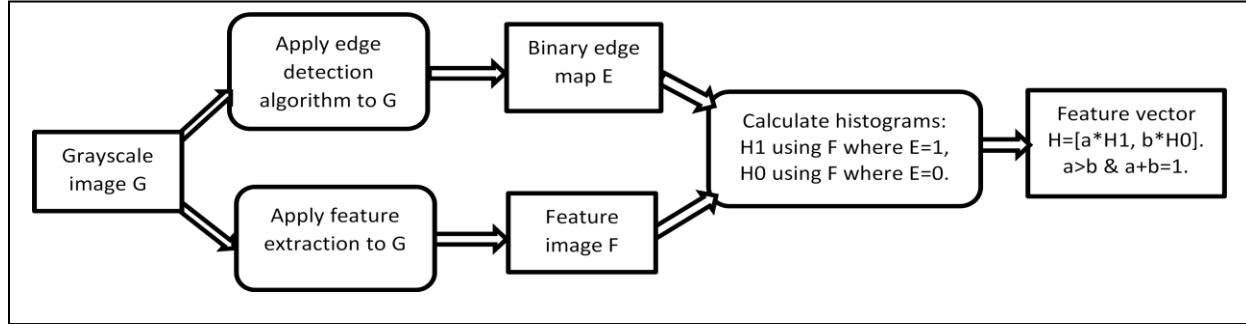


Fig.1. General algorithm for improving performance of local techniques (LBP, LTP, and LBPV)

The LBP technique uses a histogram of the LBP codes to describe the texture. This technique is characterized by its low computational complexity, resistance to lighting variations and ability to code fine details [11]

Several researchers proposed variants of the LBP coding in order to improve its performance and increase its robustness. We have included in this investigation two of these variants: the Local Ternary Pattern (LTP) and the Local Binary Patterns Variance (LBPV).

LTP [14] was introduced in an attempt to improve the discriminative power of the original operator by using three quantization levels for the intensity difference. LTP code is formally defined by (9).

$$LTP(p, r) = \sum_{k=0}^{p-1} S(I_k - I_c) * 3^k \quad (9)$$

Where

$$S(x) = \begin{cases} 1 & \text{if } x \geq \epsilon \\ -1 & \text{if } x \leq -\epsilon \\ 0 & \text{otherwise} \end{cases}$$

Because the number of resulting codes is huge (3^p), the authors suggest to use two LBP coding instead, as shown in (10) and (11).

$$LBP_high(p, r) = \sum_{k=0}^{p-1} S(I_k - I_c) * 2^k \quad (10)$$

Where

$$S(x) = \begin{cases} 1 & \text{if } x \geq \epsilon \\ 0 & \text{if } x < \epsilon \end{cases}$$

$$LBP_low(p, r) = \sum_{k=0}^{p-1} S(I_k - I_c) * 2^k \quad (11)$$

Where

$$S(x) = \begin{cases} 1 & \text{if } x \leq -\epsilon \\ 0 & \text{if } x > -\epsilon \end{cases}$$

The new coding reduces the size of the feature vector to 2^{p+1}

LBPV [16] uses a histogram that accumulates the variance values for each LBP code in attempt to improve the discriminative power of the LBP operator.

The LBPV histogram is computed as shown in (12).

$$LBPV_{p,r}(k) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} V(LBP_{p,r}(i, j), k); k = 0 \dots 2^p \quad (12)$$

Where

$$V(LBP_{p,r}(i, j), k) = \begin{cases} \text{Variance}(i, j) & \text{if } LBP_{p,r}(i, j) = k \\ 0 & \text{otherwise} \end{cases}$$

III. PROPOSED APPROACH

As mentioned in the introduction, the general approach we have adopted for improving the performance of the texture analysis techniques considered in this work, consists of separating features extracted from edge pixels/regions from those extracted from non-edge pixels/regions. The former features are then assigned higher weights. For the local techniques (LBP, LTP, and LBPV), this is achieved by first identifying edge pixels then assigning higher weights to the feature vector entries calculated from those identified pixels, see Fig1.

For the global techniques (Fourier- and wavelet-based techniques), this is achieved by identifying the regions in

the transformed space that correspond to edge pixels, and assigning higher weights to the features extracted from those regions, see Fig 2.

A. Algorithms for local techniques

The proposed approach is applied to three different local techniques, conventional LBP [10], LTP [14] and LBPV [16].

The first decision that needs to be made is to choose an adequate edge detection technique. Several edge detection operators, of different complexity and accuracy, are available in the literature. We are interested in an operator that would not slow down significantly the original technique and yet detects accurately enough the image edges; Sobel operator is chosen because it constitutes a good tradeoff between these two requirements.

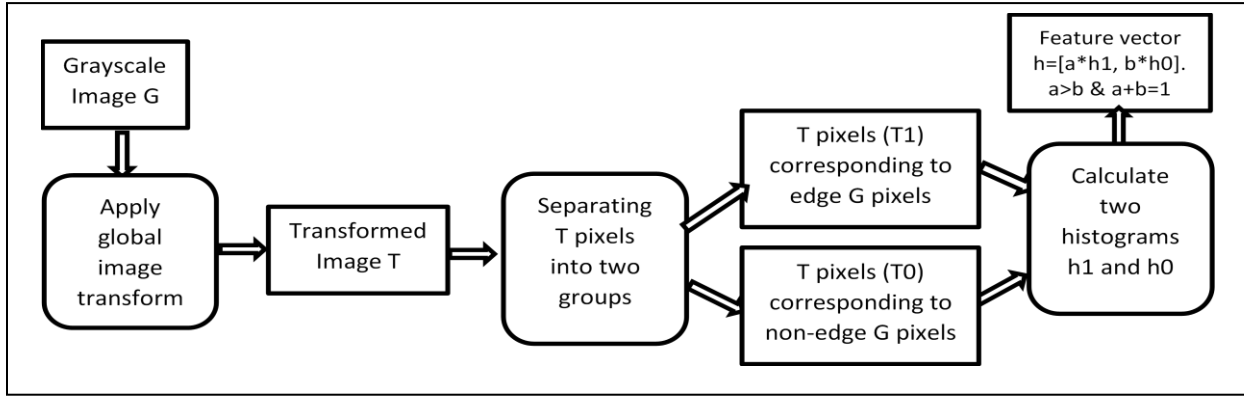


Fig.2. General algorithm for improving performance of global Techniques (FFT & DWT)

The following algorithms describe how feature vectors are calculated for the three local techniques

Algorithm 1: LBPE technique

Input: NxN grayscale image G.

Output: LBPE histogram

1. Apply LBP coding to G to obtain LBP image, LBPI.
2. Apply Sobel filter to G to calculate edge map image E (binary image, E=1 if the pixel is an edge, 0 otherwise).
3. Calculate two LBP histograms H1 and H0 using the formulas (13) and (14)

$$H1(k) = \sum_{i,j \in [1,N]} E(i,j) \cdot \delta(LBPI(i,j), k) \quad (13)$$

$$H0(k) = \sum_{i,j \in [1,N]} (1 - E(i,j)) \cdot \delta(LBPI(i,j), k) \quad (14)$$

Where

k=1 to K, where K is the number of LBP codes

$$\delta(x, y) = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{otherwise} \end{cases}$$

4. $H=[a*H1, b*H0]$; where $a+b=1$ and $a>b$.

Algorithm 2: LTPE technique

Input: NxN grayscale image G.

Output: LTPE histogram

1. Apply LTP coding to G to obtain two feature images LBP_high and LBP_low, corresponding to the two derived LBP codes.
2. Apply Sobel filter to G to calculate edge map image E (binary image, E=1 if the pixel is an edge, 0 otherwise).
3. From each feature image (LBP_high and LBP_low), calculate two histograms, one for edge pixels (H1_high and H1_low) and one for non-edge pixels (H0_high and H0_low) using the formulas (15)-(18).

$$H1_{high}(k) = \sum_{i,j \in [1,N]} E(i,j) \cdot \delta(LBP_{high}(i,j), k) \quad (15)$$

$$H1_{low}(k) = \sum_{i,j \in [1,N]} E(i,j) \cdot \delta(LBP_{low}(i,j), k) \quad (16)$$

$$H0_{high}(k) = \sum_{i,j \in [1,N]} (1 - E(i,j)) \cdot \delta(LBP_{high}(i,j), k) \quad (17)$$

$$H0_{low}(k) = \sum_{i,j \in [1,N]} (1 - E(i,j)) \cdot \delta(LBP_{low}(i,j), k) \quad (18)$$

Where

k=1 to K; K is the number of LBP codes

$$\delta(x, y) = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{otherwise} \end{cases}$$

4. H1 is a concatenation of H1_high and H1_low. H0 is a concatenation of H0_high and H0_low
5. $H=[a*H1, b*H0]$; where $a+b=1$ and $a>b$.

Algorithm 3: LBPVE technique

Input: NxN grayscale image G.

Output: LBPVE histogram

1. Apply LBP coding to G to obtain LBP image, LBPI.
2. Apply Sobel filter to G to calculate edge map image E (binary image, E=1 if the pixel is an edge, 0 otherwise).
3. Calculate local variance at each pixel in G. A variance image VAR is then obtained.
4. Feature vector H1 and H0 are calculated using (19) and (20).

$$H1(k) = \sum_{i,j \in [1,N]} E(i,j) \cdot \delta(LBPI(i,j), k) \cdot VAR(i,j) \quad (19)$$

$$H0(k) = \sum_{i,j \in [1,N]} (1 - E(i,j)) \cdot \delta(LBPI(i,j), k) \cdot VAR(i,j) \quad (20)$$

Where

k=1 to K, where K is the number of LBP codes

$$\delta(x, y) = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{otherwise} \end{cases}$$

5. $H=[a*H1, b*H0]$; where $a+b=1$ and $a>b$.

B. Algorithms for global techniques

The proposed approach is applied to two different global techniques, Discrete Fourier Transform [27] and Discrete Wavelet Transform [24]. For these techniques, there is no need to perform edge detection; instead, we segment the transformed space into low-frequency regions and high-frequency regions. The former correspond to uniform regions in the original image and the latter result mainly from edge pixels in the original image.

The wavelet transform decomposes an image into sub-bands of different resolutions; in the pyramidal wavelet decomposition that has been considered in this work, the sub-band with the lowest resolution represents the low frequencies of the image while the remaining sub-bands capture higher frequencies at different resolutions. Feature vectors extracted from the low frequency sub-band and those extracted from the rest of the sub-bands are assigned different weights and concatenated to constitute the final feature vector describing the texture image. The following algorithm summarizes this approach.

Algorithm 4: DWTE technique

Input: NxN grayscale image G.

Output: DWTE histogram

1. Apply n-level pyramidal DWT to grayscale image G. in our case $n=3$ and wavelet name is Daubechies (db4). For $n=3$, ten sub-bands are obtained: one low frequency (LL) nine high frequencies (H1,...H9)
2. Calculate the mean energy of each sub-band, $E_{LL}, E_{H1}, \dots, E_{H9}$.
3. Calculate the energy standard deviation for each sub-band, $std_{LL}, std_{H1}, \dots, std_{H9}$.
4. $H1=[E_{H1}, E_{H2}, \dots, E_{H9}, std_{H1}, std_{H2}, \dots, std_{H9}]$
5. $H0=[E_{LL}, std_{LL}]$
6. $H=[a*H1, b*H0]$; where $a+b=1$ and $a>b$

The Fourier transform of an image describes the frequency contents of that image. It is, therefore, natural to perform a segmentation of the transformed space and assign different weights to the feature vectors extracted from the different regions. In this work we have adopted the segmentation proposed in [27] which consists of splitting the Fourier space into rings representing different frequency ranges and then split the rings into wedges to obtain better discriminating features. The following algorithm summarizes the adopted approach.

Algorithm 5: FFTE technique

Input: NxN grayscale image G.

Output: FFTE Histogram

1. Apply the discrete Fourier transform to grayscale image G. FFTI is obtained.
2. Split FFTI into r rings and w wedges. In this work, $r=3$ and $w=4$. The first ring represents the low frequencies, the second ring the mid-range frequencies and the last ring the high frequencies.
3. Calculate means and standard deviations of the energy of the wedges in the low-frequency ring. A vector of $2*w$ elements is obtained, call it $r1_E$.
4. Calculate means and standard deviations of the energy of the wedges in the mid-range frequency ring. A vector of $2*w$ elements is obtained, call it $r2_E$.
5. Calculate means and standard deviations of the energy of the wedges in the high frequencies ring. A vector of $2*w$ elements is obtained, call it $r3_E$.
6. $H=[a*r1_E, b*r2_E, c*r3_E]$; $a<b$ & $a<c$; $a+b+c=1$;

C. Distance Metrics

Several similarity measurements are described in the literature. We have used the Chi2 function because it is widely adopted by texture retrieval techniques [10].

Given two texture images I and J, represented by two N-dimensional feature vectors F^I and F^J , the distance function between I and J is defined by (21):

$$D(I, J) = \text{Chi2}(F^I, F^J) = \frac{1}{2} \sum_{k=1}^N \frac{(F^I_k - F^J_k)^2}{F^I_k + F^J_k} \quad (21)$$

IV. EXPERIMENTS

We have conducted several experiments to evaluate the performance of the proposed algorithms. This section describes the datasets used in these experiments, the hardware on which they are conducted, the method adopted for evaluating the performance of the techniques and then discusses the results.

A. Testing Datasets

We have used three well-known datasets: the Brodatz collection [32], the Outex database from the University of Oulu [33], and the VisTex database from MIT Media lab [34]. Brodatz textures are photographed under controlled lighting; they are of very high quality and almost noise free. It is therefore expected to get better results with this dataset than the two other ones. Nevertheless, this dataset constitutes a good platform for testing the discriminative power of texture analysis techniques [35]. Outex textures are more challenging, since they have wide variations in terms of illumination direction and spatial resolution [35]. Vistex dataset is even tougher, because its images are taken under natural lighting conditions and they do not conform to rigid frontal plane perspectives. Therefore, they contain extra visual cues such as shadows and perspective [35]. From each dataset, images with homogeneous texture (i.e. similar texture over the whole image) have been selected. From Brodatz database, 80 images of size 640 x 640 pixels are used; from each of these images, 16 non-overlapping images of size 128x128 are extracted, which gives a database of 1216 images. From outex-TR_00000 (texture retrieval test suite), 1376 (86x16) images of size 128x128 are selected, they represent 86 different textures, and from VisTex database 928 (16x58) images of size 128x128 are chosen; they represent 58 different textures.

B. Hardware and Software Environment

We have conducted our experiments on an Intel Core 2 (2GHz) Laptop with 2 GB RAM. The software environment consists of MS Windows 7 professional and Matlab7.

C. Performance Evaluation Approach

Performance evaluation of the techniques under consideration is based on two measures: the average precision value (Pr) and the average recall value (Re). Pr is calculated as the average number of relevant texture images within the first N retrieved images (22) and (23) and Re is calculated as the average recall value of the N images in the dataset that are similar to the query image (24) and (25).

$$Pq = \frac{1}{N} \sum_{i=1}^N \delta(\phi(I_i), \phi(I_q)), \quad (22)$$

Where

$$\text{Rank}(I_i, I_q) \leq N$$

$$Pr = \frac{1}{|QDB|} \sum_{q=1}^{|QDB|} Pq \quad (23)$$

$$Rq = \frac{k}{\text{Rank}(O_k, I_q)} \quad (24)$$

$$Re = \frac{1}{N} \sum_{q=1}^N Rq \quad (25)$$

Where

Pq = precision value for query number q
 $\phi(x)$ = image which the sub - image x belongs to

$$\delta(x, y) = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{otherwise} \end{cases}$$

$\text{Rank}(I_i, I_q)$ is the rank of the similarity value of the retrieved image I_i for the query image I_q

In our experiments, N is set to 16.

|QDB| is the size of the query database. In our case, all dataset images are used once as a query, which means |QDB|=number of the images in the dataset.

O_k is k^{th} retrieved image that is similar to the query image I_q

We have also included the Precision-Recall graphs in order to visually compare the performance of the modified techniques with that of their original counterparts. The p value displayed on top of each graph gives an appreciation of the significance of the improvement. The null hypothesis that is tested is the following:

H0: "There is no significant difference between the performance of the original method and that of the modified method".

We have applied a one-sample t-test (since the sample size is relatively small; it consists of the 16 values that have been used to draw the precision-recall graphs). And the results shown are for a significance level $\alpha = 0.05$.

D. Experiment 1: Identification of the Optimal Weights

The first step in the implementation of the proposed approach is to identify, for each technique, the optimal set of weights to be associated with the edge and non-edge feature vectors. For each technique, we have compared its retrieval performance on the Brodatz dataset for different sets of weights. The one producing the best performance is adopted for the remaining experiments. Table 1 and Table 2 show the results obtained for each technique and highlights the highest retrieval performances as well as the sets of thresholds producing them.

Table 1. Identification of the optimal set of weights for LBPE, LTPE, LBPVE and DWTE

Performance Weights(a, b)	LBPE P (%)	LTPE P (%)	LBPVE P (%)	DWTE P (%)
(0.1, 0.9)	89.50	93.02	84.72	61.65
(0.2, 0.8)	90.25	93.29	86.20	68.90
(0.3, 0.7)	90.71	93.55	87.18	72.85
(0.4, 0.6)	91.19	93.89	87.98	75.04
(0.5, 0.5)	91.56	94.16	88.53	76.74
(0.6, 0.4)	91.83	94.45	88.93	78.11
(0.7, 0.3)	92.00	94.77	89.01	79.36
(0.8, 0.2)	91.84	94.95	88.68	80.36
(0.9, 0.1)	90.74	94.39	87.45	80.78
(0.95, 0.05)	-	-	-	79.77
(1.0, 0.0)	0.00	0.00	75.95	69.82
(0.0, 1.0)	88.86	92.69	82.95	11.72

Table 2. Identification of the optimal set of weights for FFTE

Performance Weights (a, b)	FFTE P (%)
(0.8, 0.2, 0.0)	77.05
(0.6, 0.2, 0.2)	80.19
(0.4, 0.2, 0.4)	82.58
(0.2, 0.2, 0.6)	84.876
(0.0, 0.2, 0.8)	72.98
(0.6, 0.4, 0.0)	80.42
(0.4, 0.4, 0.2)	82.99
(0.2, 0.4, 0.4)	85.10
(0.0, 0.4, 0.6)	72.95
(0.4, 0.6, 0.0)	82.66
(0.2, 0.6, 0.2)	84.88
(0.0, 0.6, 0.4)	71.50
(0.2, 0.8, 0.0)	84.13
(0.1, 0.8, 0.1)	84.79
(0.0, 0.8, 0.2)	69.28
(0.0, 1.0, 0.0)	63.97
(1.0, 0.0, 0.0)	66.98
(0.8, 0.0, 0.2)	74.79
(0.6, 0.0, 0.4)	78.46
(0.4, 0.0, 0.6)	81.18
(0.2, 0.0, 0.8)	83.30
(0.0, 0.0, 1.0)	55.54

E. Discussion of Experiment 1 Results

All the results shown in Table 1 and Table 2 indicate that the optimal results are obtained when the feature vectors derived from edge regions are given higher weights than those derived from uniform regions ($a=0.7$ to 0.9). These results corroborate with the research findings indicating that edge regions carry more discriminative information than uniform regions. Which may explain why the human eyes focus on these regions when looking at objects as mentioned in [5]. The tables also show that the performance drops sharply when only the features extracted from the edge regions are used. This clearly indicates that uniform regions still carry useful information needed for texture characterization.

F. Experiment 2: Evaluation of the Effectiveness and Efficiency of Proposed Algorithms

We have conducted several experiments to evaluate the effectiveness of the proposed algorithms. For each technique considered in this work (LBP, LTP, LBPV, DWT and FFT) and for each dataset (BRODATZ, OUTEX, and VISTEX), the average precision (Pr), as well as the average extraction and matching times (E and M), are calculated for both original technique and modified one. The results are reported in the three tables (Table 3 – Table 5). Resulting precision-recall graphs as well as calculated p-values, are show in Fig 3 - Fig 5.

G. Discussion of Experiment 2 Results

Several observations about the effectiveness and efficiency of the proposed approach can be made from the results reported in the aforementioned tables and figures.

First, we notice that for the three datasets considered in this work, the modifications introduced by our approach to the original techniques always lead to an improvement in their performance; and that, the occurring improvement is always proven to be statistically significant (low p-values). These combined results constitute a strong evidence of the effectiveness of the proposed algorithms.

Second, we notice also, that the overall performance of the techniques under investigation is not the same on the three datasets. The highest average performance is obtained on Brodatz (average performance of the original methods = 84.04% and average performance of the modified methods = 88.37%), and the lowest average performance is obtained on Vistex (70.28% for the original methods and 75.03% for the modified ones). This is simply due to the fact that Brodatz dataset is the less challenging among the three considered in the experiments while Vistex is the one that incorporates more challenges as described in section (A. Testing Datasets).

Our third observation is about the amount of improvement introduced by the approach; we can notice that the proposed approach leads to more improvement for Vistex (average of 4.75%) and Brodatz (average of 4.33%) than for Outex (average of 2.72%). This is probably due to the nature of the images in the three datasets. Edges in the Outex images used in the experiment happen to be less prominent than those of the images found in Brodatz and Vistex sets respectively. This suggests that the proposed approach is more effective for images with prominent edges.

The improvement in the retrieval rate obtained by the proposed approach comes with an increase in the processing time that may occur at two different stages: when the feature vector is calculated and when the query and dataset feature vectors are compared. Calculation of the new feature vector requires identification of the edge regions and the non-edge regions. This operation requires more time for the local techniques because they require an

explicit edge processing (Sobel filtering). For global techniques, this operation is relatively fast since the process consists of simple thresholding operations. The increase in the execution time of the feature vector comparison occurs when the size of the feature vector increases. Again, this is more significant on local techniques than on global ones, because the size of the feature vectors of the latter do not change drastically.

The results obtained in experiment 2, demonstrate both the effectiveness and the Efficiency of the proposed method.

H. Experiment 3: Evaluation of the Robustness of the Proposed Algorithms to Noise

It is well known that edge calculation is, in general, sensitive to noise. One of the reasons is the fact that both edges and noise are in the high-frequency range. Therefore, it is legitimate to check whether the proposed approach weakens the robustness of the original techniques in a noisy environment. To do so, we have applied a Gaussian noise to BRODATZ dataset images and compared the results of the original methods with those of the modified techniques. In the experiment, we applied a Gaussian noise with mean=0 and STD=0.01. The results are shown in Table 6.

I. Discussion of Experiment 3 Results

By comparing the results shown in Table 6 with those shown in Table 3, we notice that the performance of all the techniques have deteriorated but the gaps between the performance of the original techniques and that of the modified ones have significantly increased (2.48% against 4.64% for LBP, 1.66% against 2.56% for LTP, 3.49% against 5.07% for LBPV, 6.20% against 8.02% for FFT, and 5.15% against 8.88% for DWT). This indicates clearly

that the proposed approach strengthens the resistance of the techniques to noise.

V. CONCLUSION

There are several research works in the field of physiology suggesting that edge regions get more attention from the visual system of primates in general and humans in particular when analyzing scenes. Based on that suggestion, we proposed an approach for improving the performance of several existing texture analysis techniques. The approach is based on a simple idea consisting of separating edge regions/pixels from non-edge regions/pixels and extracting the features from both regions separately. The final feature vector is then a weighted combination of the two types of feature vectors. Where those extracted from edge regions are given higher weights. To check the effectiveness of this approach, we have devised five different algorithms that implement the proposed approach on five selected techniques (three local and two global techniques). The experiments we have conducted on three popular datasets showed that the implementation of the proposed approach significantly improves the performance of the tested techniques. They also showed that the required extra execution time is very negligible for global techniques and reasonable for the local ones. Finally, the experiments have showed that the proposed approach improves even the robustness of the technique against noise.

We are planning to apply the proposed approach to a larger list of techniques and explore the means for extending its implementation on existing texture classification techniques.

Table 3. Summary of the results obtained on BRODATZ dataset Note that, in tables 2-5, FFT and FFTE represent the original and modified Fourier based techniques respectively. DWT and DWTE represent the original and modified Discrete Wavelet Transform respectively. P (%) represents the average precision as defined by equations (2 & 3). E (s) and M (s) represent the average times, in second, for feature extraction and matching respectively

BRODATZ Dataset										
	LBP	LBPE	LTP	LTPE	LBPV	LBPVE	FFT	FFTE	DWT	DWTE
P (%)	89.52	92.00	93.29	94.95	85.52	89.01	78.90	85.10	72.95	80.78
Diff (%)	2.48		1.66		3.49		6.20		7.83	
E(s)	0.003	0.008	0.006	0.010	0.010	0.029	0.002	0.003	0.004	0.004
M(s)	0.003	0.004	0.006	0.008	0.003	0.004	0.003	0.003	0.002	0.002

Table 4. Summary of the results obtained on OUTEX dataset.

OUTEX Dataset										
	LBP	LBPE	LTP	LTPE	LBPV	LBPVE	FFT	FFTE	DWT	DWTE
P (%)	84.43	86.80	87.73	89.15	83.24	86.00	71.38	76.98	72.37	73.81
Diff (%)	2.37		1.42		2.76		5.60		1.44	
E(s)	0.003	0.008	0.005	0.010	0.010	0.029	0.003	0.003	0.004	0.004
M(s)	0.004	0.004	0.006	0.009	0.004	0.004	0.003	0.003	0.002	0.002

Table 5. Summary of the results obtained VISTEX dataset.

VISTEX Dataset										
	LBP	LBPE	LTP	LTPE	LBPV	LBPVE	FFT	FFTE	DWT	DWTE
P (%)	75.28	77.89	77.42	79.72	69.04	73.28	66.24	73.55	63.41	70.70
Diff (%)	2.61		2.30		4.24		7.31		7.29	
E(s)	0.003	0.008	0.005	0.010	0.010	0.030	0.002	0.003	0.004	0.004
M(s)	0.002	0.003	0.004	0.006	0.002	0.003	0.003	0.003	0.002	0.002

Table 6. Comparison of average precisions of the original techniques and modified ones, obtained on noisy BRODATZ dataset.

Noisy BRODATZ Dataset										
	LBP	LBPE	LTP	LTPE	LBPV	LBPVE	FMS	FMSE	DWT	DWTE
P (%)	80.89	85.53	88.01	90.57	78.85	83.92	75.09	83.11	67.62	76.50
Diff (%)	4.64		2.56		5.07		8.02		8.88	

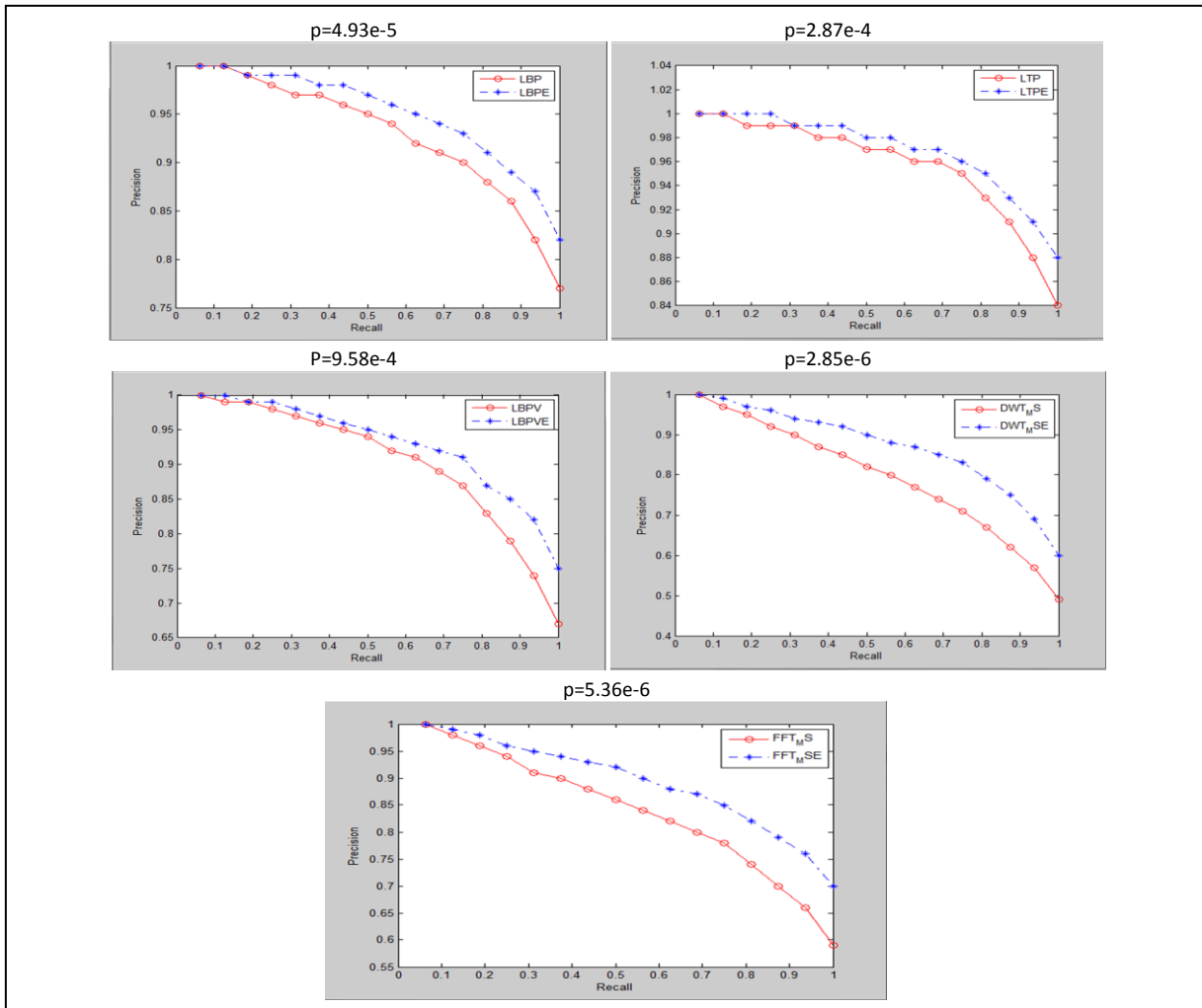


Fig.3. Precision-Recall graphs of the original and modified techniques obtained on BRODATZ

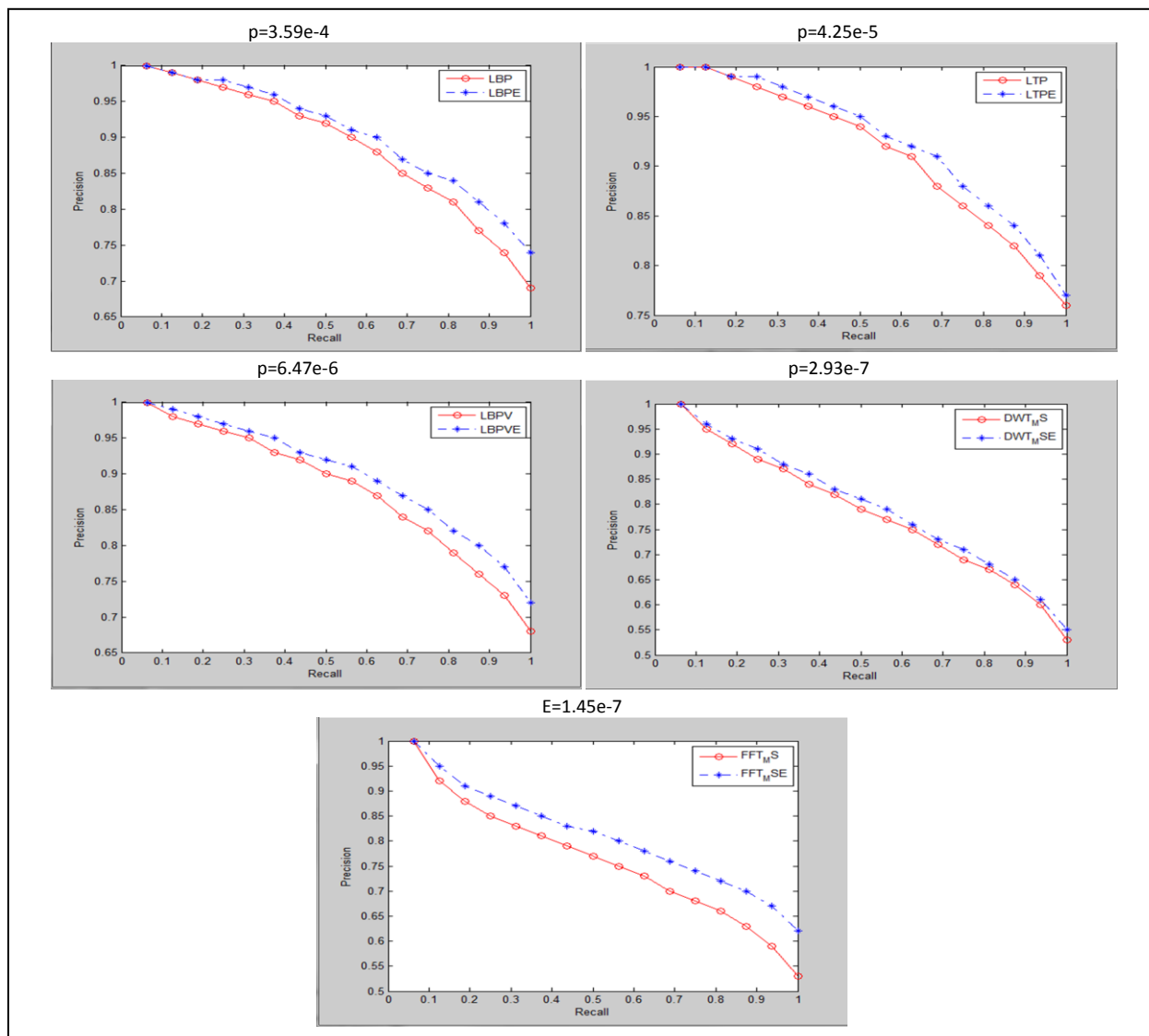


Fig.4. Precision-Recall graphs of the original and modified techniques obtained on OUTEX

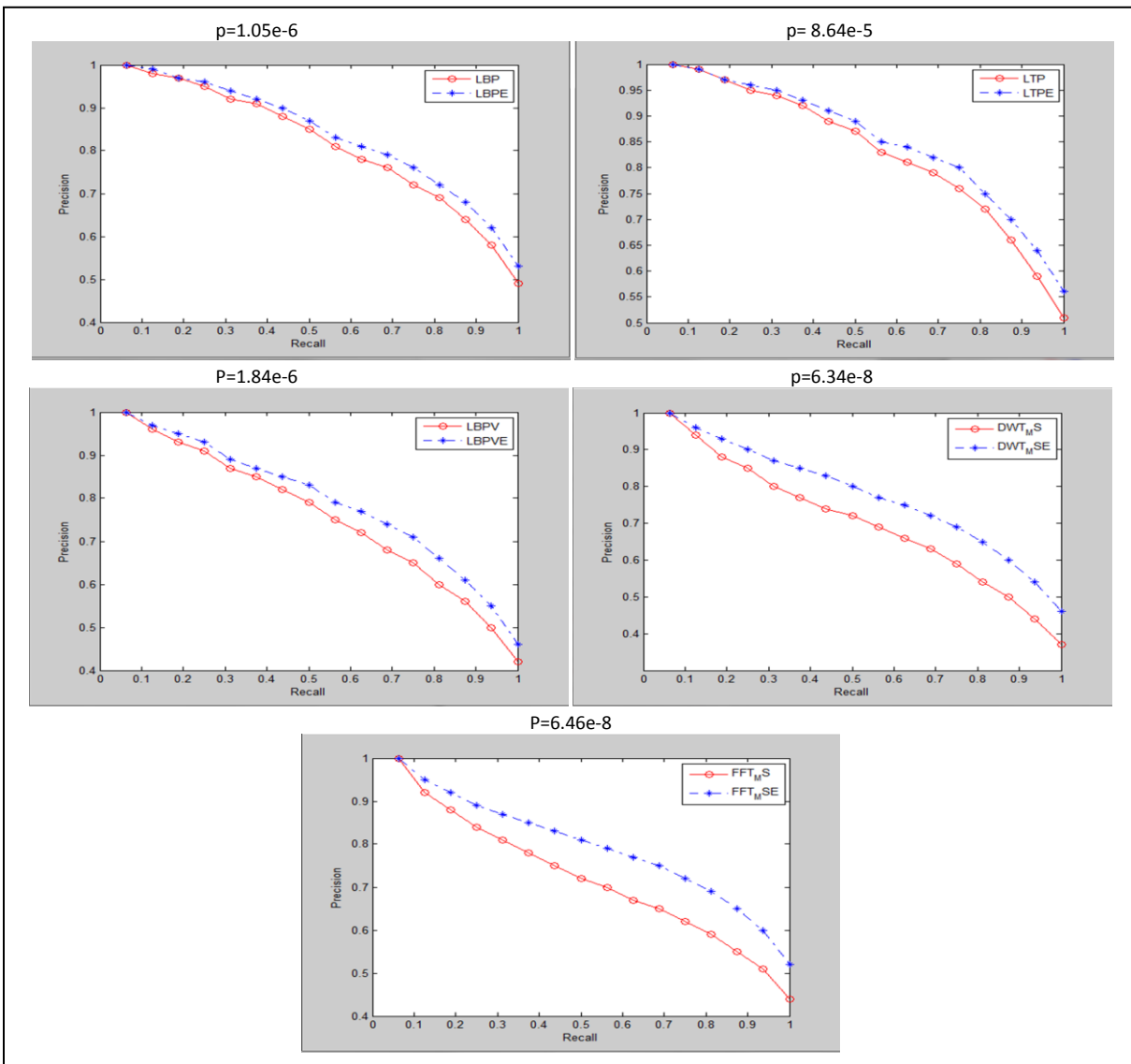


Fig.5. Precision-Recall graphs of the original and modified techniques obtained on VISTEX

REFERENCES

- [1] R. M. Shapley and D. J. Tolhurst, Edge Detectors in Human Vision, *The Journal of Physiology* vol. 229, pp. 165-183, 1973
- [2] Marr, D., 1982. *Vision*. W.H. Freeman and Co., New York.
- [3] M. C. Morrone and D. C. Burr, Feature Detection in Human Vision, *Proc. of the Royal Society of London*, B235, 221-245, 1988
- [4] B.W. Tatler, R.J. Baddeley, I.D. Gilchrist, Visual correlates of fixation selection: effects of scale and time, *Vision Research*, vol. 45, pp. 643-659, March 2005.
- [5] R. J. Baddeley, & B.W. Tatler, High frequency edges (but not contrast) predict where we fixate: A Bayesian system identification analysis, *Vision Research*, vol. 46, pp. 2824-2833, September 2006.
- [6] Mihran Tuceryan, *Texture Analysis, the Handbook of Pattern Recognition and Computer Vision* (2nd Edition), pp 207-248, World Scientific Publishing Co., 1998.
- [7] L. Shapiro and G. C. Stockman, *Computer Vision*, chapter 7, pp 212-225, Prentice Hall, 2001.
- [8] B. Zhang, Y. Gao, S. Zhao and J. Liu, Local derivative pattern versus local binary pattern: Face recognition with high-order local pattern descriptor, *IEEE Trans. On Image Processing*, vol. 19, pp. 533-544, February 2010.
- [9] U S N Raju, A Sridhar Kumar, B Mahesh, B Eswara Reddy, *Texture Classification With High Order Local Pattern Descriptor: Local Derivative Pattern*, *Global Journal of Computer Science and Technology*, Vol. 10 Issue 8, pp September 2010
- [10] T. Ojala, M. Pietikäinen, and D. Harwood, A Comparative study of texture measures with classification based on feature distributions. *Pattern Recognition*. 29, 51-59, 1996.

- [11] L. Nanni, A. Lumini, and S. Brahmam, Survey on LBP based texture descriptors for image classification, *Expert System With Applications*, 39, 3634-3641, 2012.
- [12] T. Ojala, M. Pietikäinen and T. Mäenpää, Gray scale and rotation invariant texture classification with local binary patterns. In the 6th European Conference on Computer Vision-Part I. Springer-Verlag London, UK, 2000.
- [13] T. Ojala, M. Pietikäinen, and Topi Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 24, 971-987, 2002.
- [14] X. Tan and B. Triggs, Enhanced Local Texture Feature Sets for Face Recognition Under Difficult Lighting Conditions, *Analysis and Modeling of Faces and Gestures (AMFG'07)*, LNCS, 4778, 168-182, Springer 2007.
- [15] Z. Guo, L. Zhang and D. Zhang, A Completed Modeling of Local Binary Pattern Operator for Texture Classification, *IEEE Trans. On Image Processing*, 19, 1657-1663, 2010.
- [16] Z. Guo, L. Zhang and D. Zhang, Rotation Invariant Texture Classification Using LPB Variance (LBPV) with Global Matching, *Pattern Recognition*, 43, 706-719, 2010.
- [17] W. Li, and K. Mao, Designing compact Gabor filter banks for efficient texture feature extraction, 11th International Conference on Control Automation Robotics & Vision (ICARCV), Singapore, 2010
- [18] M. M. H. Daisy, S.T. Selvi, and J. S. G. Mol, Combined texture and shape features for content based image retrieval International Conference on Circuits, Power and Computing Technologies (ICPCT), Nagercoil, 2013
- [19] M.P. Arakeri, and G.R.M. Reddy, A novel CBIR Approach to differential Diagnosis of Liver Tumor on Computed Tomography Images, International Conference on Modeling, Optimization and Computing, *Procedia Engineering*, 38, 528-536, 2012.
- [20] H-J. Wang, H-N. Qi and X-F Wang, A new Gabor based approach for wood recognition, *Neurocomputing* 116 192-200, 2013
- [21] J.R. Smith and S-F. Transform features for texture classification and discrimination in large image databases, *Image processing proceedings, ICIP-94*, 3, 407-411 1994.
- [22] M. Kokare, P.K. Biswas, B.N. Chatterji, Texture image retrieval using rotated wavelet filters, *Pattern Recognition Letters*, 28, 1240-1249, 2007
- [23] P.W. Huang, S.K. Dai, Image retrieval by texture similarity, *Pattern Recognition*, 36, 665-679, 2003.
- [24] P.W. Huang, S.K. Dai, Design of a two-stage content-based image retrieval system using texture similarity, *Information Processing and Management* 40, 81-96, 2004.
- [25] P.W. Huang, S.K. Dai, and P.L. Lin, Texture image retrieval and image segmentation using composite sub-band gradient vectors, *J. Vis. Communication and Image Representation* 17, 947-957, 2006
- [26] K. Arai, I. N. Abdullah, H. Okumura, Image Retrieval Based on Color, Shape, and Texture for Ornamental Leaf with Medicinal Functionality Images, *IJIGSP*, vol.6, no.7, pp.10-18, 2014
- [27] D-M. Tsai and C-F Tseng, Surface roughness classification for castings, *Pattern Recognition* 32, 389-405, 1999.
- [28] J.S. Weszka, C.R. Dyer, A. Rosenfeld, A comparative study of texture measures for terrain classification, *IEEE Trans. System, man and Cybernetics*, 6, 269-285, 1976.
- [29] A. Abdesselam, Multi-resolution Fourier-based texture image retrieval, *Proceedings of the 2009 conference on Information Science, Technology and Applications*, pp 72-77, ACM Digital Library
- [30] Beck et al, Spatial frequency channels and perceptual grouping in texture segregation, *Comp. Vision Graphics and Image Processing*, 37, 299-325, 1987
- [31] A. Abdesselam, Improving Local Binary Patterns Techniques by Using Edge Information, *Lecture Notes on Software Engineering* vol. 1, no. 4, pp. 360-363, 2013
- [32] <http://www.ux.uis.no/~tranden/brodatz.html>, accessed in 2009.
- [33] <http://www.outex.oulu.fi/index.php?> accessed in 2012.
- [34] <http://vismod.media.mit.edu/pub/VisTex>, accessed in 2014
- [35] S. Hossain and S. Serikawa, Texture databases – A comprehensive survey, *Pattern Recognition Letters*, 34, 2007-2022, 2013.

Authors' Profiles



Asst. Professor of Computer Science. His research interests include content-based image retrieval, biometrics, medical image analysis and machine learning.

Dr. Abdelhamid Abdesselam received his MSc. From Université Paul Sabatier, Toulouse, France and his PhD in Computer Vision from Ecole National Polytechnique de Toulouse, France. He worked in Universiti Malaysia Sarawak and Qatar University before joining Sultan Qaboos University. He is currently an

How to cite this paper: Abdelhamid Abdesselam, "Edge Information for Boosting Discriminating Power of Texture Retrieval Techniques", *International Journal of Image, Graphics and Signal Processing(IJIGSP)*, Vol.8, No.4, pp.16-28, 2016. DOI: 10.5815/ijigsp.2016.04.03