Dominant Local Feature based Rotation Invariant Texture Classification

J. Subha,
PG Scholar,CSE,
GCE,Tirunelveli.

G.Tamil Pavai,M.E,
Assistant Professor (Sr.Gr),
CSE, GCE, Tirunelveli.

S.Tamil Selvi,
Professor, ECE,
NEC, Kovilpatti.

ABSTRACT
This paper proposes a new approach for Dominant Local Feature Based Rotation Invariant Texture Classification. Texture classification plays an important role in computer vision and image processing applications. The applications include medical image analysis, understanding, remote sensing, object based image coding and image retrieval. As the demand of increase in applications, texture classification has received considerable attention over the last several decades and numerous novel methods have been proposed. The proposed approach extracts the features with dominant local binary patterns (DLBP) in a texture image. The conventional LBP approach is extended to the dominant local binary pattern (DLBP) approach in order to effectively capture the dominating patterns in texture images. Unlike the conventional LBP approach, which only exploits the uniform LBP, given a texture image, the DLBP approach computes the occurance frequencies of all rotation invariant patterns defined in the LBP groups. These patterns are then sorted in descending order. The first several most frequently occurring patterns should contain dominating patterns in the image and those patterns only taken for the classification. Therefore, are the dominant patterns. It is shown that DLBP approach is more reliable to represent the dominating pattern information in the texture images. Using this developed approach several testings have been done. For testing purpose Brodatz, Outex images, have been used and compared with six published texture features in the experiments. It is experimentally demonstrated that the proposed method achieves the highest classification accuracy in texture data bases and image conditions.

Index Terms
Dominant Local Binary Pattern, Texture classification, Gabor filter.

1. INTRODUCTION
TEXTURE can be used in the analysis of images in several ways: in the segmentation of scenes into distinct objects and regions, in the classification or recognition of surface materials, and in the computation of surface shape. Textures are replications, symmetries and combinations of various basic patterns or local functions, usually with some random variation. This means that they are particularly useful for searching visual databases and other human computer interaction applications. However, since the notion of texture is spatial variations of intensity or color. S.Liao et al. applied the Dominant Local Binary Pattern (DLBP) is the most frequently occurred patterns in a texture image. It is shown that DLBP approach is more reliable to represent the dominant patterns[1]. H. Arof et al. presents a texture descriptor that utilises circular neighbourhoods with 1-D discrete Fourier transforms to obtain rotation-invariant features [2]. A.C.Bovik et al. used computational approach for analyzing visible textures is described. Textures are modeled as irradiance patterns containing a limited range of spatial frequencies, where mutually distinct textures differ significantly in their dominant characterizing frequencies [3]. R. Chellappa et al. present the least square (LS) estimates of model parameters are used as feature extraction methods for the classification of textures using two-dimensional (2-D) Markov random field (MRF) models [4]. Huawu Deng et al demonstrated an isotropic circular Gaussian MRF (ACGMRF) model for retrieving rotation-invariant texture features. To overcome the singularity problem of the least squares estimate method, an approximate least squares estimate method is designed and implemented, Rotation-invariant features are obtained from the ACGMRF model parameters using the discrete Fourier transform [5].George M. Haley et al proposed rotation-invariant texture classification based on a complete space-frequency model. A polar, analytic form of a two-dimensional (2-D) Gabor wavelet is developed, and a multiresolution family of these wavelets is used to compute information-conserving microfeatures [6]. X. Huang et al. extended for analyzing the features. The independent component feature method derives first a Gabor feature vector based upon a set of down-sampled Gabor wavelet representation of images by different orientations and scale local features. Gabor feature vector dimensionality has been reduced by principal component analysis and derives Independent Gabor Features. For classification Principal Radon Method is used. Gabor transformed the characteristics of spatial locality, Scale and orientation [7].

L.M.Kaplan proposed the effectiveness of multiscale Hurst parameters as features for texture classification and segmentation. For texture classification, the performance of the generalized Hurst features is compared to traditional Hurst and Gabor features [8]. Rangasami et al. Introduces the texture classification which is rotation invariant, i.e., the recognition accuracy is not affected if the orientation of the test texture is different from the orientation of the training samples [9]. Shu-taoli et al. investigate the texture classification problem with individual and combined multiresolution features, i.e., dyadic wavelet, wavelet frame, Gabor wavelet, and steerable pyramid. The support vector machines are used as classifiers it show that the steerable pyramid and Gabor wavelet classify, texture images with the highest accuracy [10]. Mallat introduced the Multiresolution representations are very effective for analyzing the information content of images. Which generates coefficients in the HL, LH, and LL channels for subsequent classification tasks the application of this representation to data compression in image coding, texture discrimination and fractal analysis [11]. T. Ojala et al [12][17][18]presents a theoretically very simple, yet efficient, multiresolution approach to gray-scale and rotation invariant texture classification based on local binary patterns and nonparametric discrimination of sample and prototype distributions. The method is based on recognizing that certain local binary patterns, termed "uniform" are fundamental properties of local image texture and their occurrence histogram is proven to be a very powerful texture feature it characterize the spatial configuration of local image texture and the performance can be further improved by combining them with rotation invariant variance measures that characterize the contrast of local image texture, with a pattern recognition approach to accurate camera-based color measurements on 3-dimensional histograms. R. Porter et al.[13] presents the wavelet transform, a circularly symmetric Gabor filter or a Gaussian Markov random field with a circular neighbour set to achieve rotation-invariant texture classification. H.Yao et al.[14] introduces the Correlation of the similar textures,omits the effect of unimportant pixels with sparse distribution. Experimental results approve that the novel method do achieve a promotion performance in remote sensing image retrieval based on the classification of Gabor texture features. J. Zhang et al. [15] introduces the circular Gabor filters (CGF) for rotation invariant texture segmentation. By the circular symmetric version the rotation invariant texture features are achieved via the channel output of the circular Gabor filters.

2. DOMINANT LOCAL BINARY PATTERN (DLBP)

2.1 Overview of Local Binary Patterns (LBP)

Local binary patterns (LBP) represent the state-of-the-art for a wide range of detection problem. In particular, detectors exploiting LBPs have achieved highly competitive results in areas including texture and dynamic texture classification. The original LBP operator was introduced by Ojala et al. [12], and was proved a powerful means of texture description. The fig :1 shows the input Brodatz texture imageD6 (wire) and the corresponding LBP image with the LBP histogram. The operator labels the pixels of an image by thresholding a 3 x3 neighborhood of each pixel with the center value and considering the results as a binary number and the 256-bin histogram of the LBP labels computed over a region is used as a texture descriptor.The derived binary numbers (called Local Binary Patterns or LBP codes) codify local primitives including different types of curved edges, spots, flat areas, etc , The first creature of the operator worked with the eight-neighbors of a pixel, using the value of the average of the block considered to be a center pixel as a threshold.

$$LBP_{m,R} = \min_{0 \leq n < m} \{ \sum_{i=0}^{m-1} u(ti - tc) 2^{[(i+n) mod\ m]} \} \rightarrow (1)$$

Where tc-centre of the neighboring pixel, ti - ith neighboring pixel, i=0,….m-1, n- total pixel for an input image, m-total number of neighboring pixels, r - radius of the circle which determines how far the neighboring pixels are located away from the center pixel, and u(x)=1 if x>0 else u(x)=0. The value m is assigned according to the value of R as suggested in[12].The LBP label for that center pixel is given by (1)where represents the center pixel, is the neighboring pixel, is the total number of neighboring pixels, is the circle radius which determines how far the neighboring pixels are located away from the center pixel, and if else . The value of is assigned according to the value Furthermore, the absolute pixel intensity information at and is discarded by using the step function in (1) when calculating LBP.

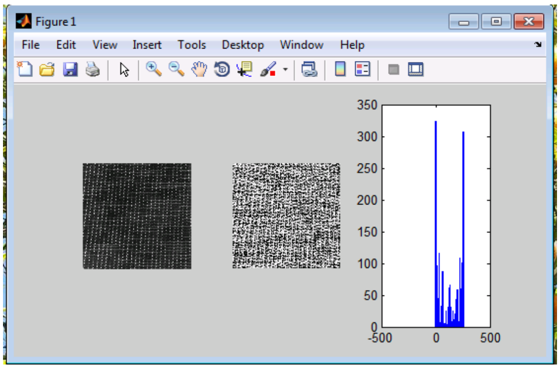

Fig 1: Example of an input image, the corresponding LBP image and histogram.

Therefore, the LBP operator is not sensitive to histogram equalization. An LBP code for a neighborhood was produced by multiplying the threshold values with weights given to the corresponding pixels, and summing up the result Fig. 2(a). Since the LBP was, by definition, invariant to monotonic changes in gray scale, it was supplemented by an independent measure of local contrast. The average of the gray levels below the center pixel is subtracted from that of the gray levels above (or equal to) the center pixel. The method using local binary patterns (LBPs) is to encode the pixel-wise information in the texture images. At the average of the center pixel, each neighboring pixel is assigned with a binary label, which can be either "0"or "1," depending on whether the center pixel has higher intensity value than the neighboring pixel (Fig. 2(a) for an illustration). The neighboring pixels are the angularly evenly distributed sample points over a circle with radius centered at the center pixel.In LBP, The comparison operator between single pixels in LBP is simply replaced with comparison between average gray-values of sub-regions Fig:2(b) Each sub-region is a square block containing neighboring pixels (or just one pixel particularly). The whole filter is composed of 9 blocks. We take size's' of the filter as a parameter, and s x s denoting the scale of the LBP operator (particularly, 3 x 3 LBP is in fact the original LBP).A Local Binary Pattern is called uniform if it contains at most two bitwise transitions from 0 to 1 or vice versa when the binary string is considered circular. For example, 00000000, 001110000 and 11100001 are uniform patterns. To create an LBP representation an input texture image must first be converted to greyscale before this operator is applied to each individual pixel within the image. A feature vector describing the textural properties of the image is then obtained from a histogram of the LBP values of the image. In this work, we represent's texture analysis based on Local Binary Pattern (LBP) features The most important properties of LBP features are their tolerance against illumination changes and their computational simplicity. We examine the, Support Vector Machine (SVM), to perform texture classification using LBP features. The support vector machine is a theoretically superior machine learning methodology with great in classification of high dimensional datasets and has been found competitive with the best machine learning algorithms.

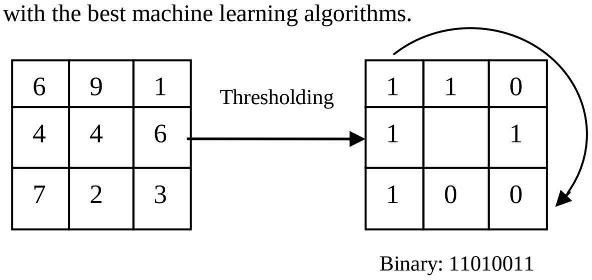

Fig 2(a): The basic LBP operator.

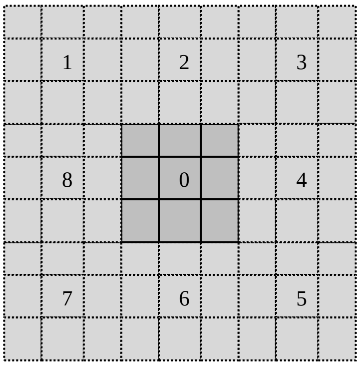

Fig 2(b):The 9 x 9 LBP operator.

SVMs have been tested and evaluated only as pixel-based image classifiers The SVM method was designed to be applied only for two class problems[1][10]. For applying SVM to multi-class classifications, two main approaches have been suggested. The basic idea is to reduce the multi-class to a set of binary problems so that the SVM approach can be used. For comparison purposes, an object-based classification of the same image was performed. The training samples in both cases were the same and were obtained and evaluated.

2.2. Dominant Local Binary Patterns (DLBP)

In the conventional LBP method only the uniform LBPs are considered. At a pixel, it gives a uniform LBP if the corresponding binary label sequence has no more than two transitions between "0" and"1" among all pairs of the adjacent binary labels. For example, the binary label sequences "10001111" and "00011000" are uniform LBPs. But the sequence "01001111" is not a uniform LBP because it has four transitions. In the textures which mostly consist of straight edges or low curvature edges, the uniform LBPs effectively capture the fundamental information of textures. Although utilizing the uniform LBPs is insufficient to capture textural information, it avoids considering all the possible patterns to perform classification. The occurrence frequencies of different patterns vary greatly and some of the patterns rarely occur in a texture image. The proportions of these patterns are too small and inadequate to provide a reliable estimation of the occurrence possibilities of these patterns. Therefore, we propose to use dominant local binary patterns (DLBPs) [1] which consider the most frequently occurred patterns in a texture image. It avoids the aforementioned problems encountered by merely using the uniform LBPs or making use of all the possible patterns, as the DLBPs are defined to be the most frequently occurred patterns. It will be demonstrated that a minimum set of pattern labels that represents around 80% of the total pattern occurrences in an image can effectively captures the image textural information for classification tasks.

The DLBPs are defined to be the most frequently occurred patterns. This required number of patterns remains the same as the DLBP features are subsequently extracted from the training image set or new testing images. Nonetheless, for two different texture images, the dominant patterns can be of different types. That is, the DLBP approach is not limited to consider only a fixed set of patterns (e.g., uniform patterns). This is distinct to the conventional LBP framework, in which the final feature vector representing an input image is the occurrence histogram of the fixed set of uniform patterns. To retrieve the DLBP feature vectors from an input image, the pattern histogram which considers all the patterns in the input image is constructed and the histogram bins are sorted in non increasing order. Based on the previously computed number of patterns, the occurrence frequencies corresponding to the most frequently occurred patterns in the input image are served as the feature vectors. It is noted that the DLBP feature vectors do not bear information regarding the dominant pattern types, and they only contain the information about the pattern occurrence frequencies. According to the experimental results, omitting the dominant pattern type information in the DLBP feature vectors is not harmful. It is because the 80% dominant patterns of DLBP is a superiority of the overall patterns, and the DLBP feature vectors are already very descriptive. Without encapsulating the pattern type information, the DLBP features also possess surpassing robustness against image noise, as compared to the conventional LBP features. Under the effect of image noise, the binary label of a neighboring pixel is possible to be flipped by the intensity distortion induced by noise. Flipped binary labels alter the extracted LBPs. As a result, even though some LBPs are computed on the same type of image structures, the extracted LBP type can vary significantly. Thus, the pattern type information is unreliable. In the conventional LBP framework, the pattern types are categorized as uniform patterns or non uniform patterns. In which, under the effect of image noise, a large amount of useful patterns turns into non uniform ones that are unconsidered in the conventional LBP method. On the contrary, the DLBP approach processes all 80% dominant patterns disregarding the pattern types. The DLBP approach is then capable of capturing more pattern types to deal with the pattern distortion.

Input: Training image set
step1: Initialize k temp
step2: Read tained dataset
step3: Initialize the histogram size
step4: compute the pixel-wise LBP training set
step5: update the histogram according to LBP
step6: sort the histogram in descending order
step7: find the no. of patterns of 80% occurances as K
step8: Is image occurance train the dataset
step9:The required no. of patterns of 80% occurances.

3. GABOR FILTER

Gabor filter is a impulse response defined by a harmonic function multiplied by a Gaussian function. Because of the multiplication-convolution property (Convolution theorem), the Fourier transform of a Gabor filter's impulse response is the convolution of the Fourier transform of the harmonic function and the Fourier transform of the Gaussian function. The filter has a real and an imaginary component representing orthogonal directions.The two components may be formed into a complex number or used individually, Gabor filters are directly related to Gabor wavelets, since they can be designed for a number of dilations and rotations [5]. However, in general, expansion is not applied for Gabor wavelets, since this requires computation of bi-orthogonal wavelets, which may be very time-consuming. Therefore, usually, a filter bank consisting of Gabor filters with various scales and rotations is created. The filters are convolved with the signal, resulting in a so-called Gabor space. The real part of the complex Gabor function is a good fit to the receptive field weight functions found in simple cells.

$$\mathrm{Gi}(\mathrm{x},\mathrm{y}) = \frac{1}{2*pi*sx*sy} * \exp\left(\left[\frac{-1}{2}\left\{\frac{x^2}{sx}+\frac{y^2}{sy}2\right\}\right]\right) * \mathrm{Mi}(\mathrm{x},\mathrm{y},\mathrm{f})$$

M1(x,y,f) = cos[2*pi*f*sqrt(x^2+^{y2})];

M2(x,y,f) = cos[2*pi*f*(x*cos(theta)+y*sin(theta)]

(1)I: Input image
(2)Sx & Sy:Variances along x and y-axes respectively
(3) f: The frequency of the sinusoidal function
(4) theta:The orientation of Gabor filter
(5)G1 & G2: The output filters as described above
(6) gabout 1 & gabout 2 :The output filtered images.

The Gabor space is very useful in image processing applications such as optical character recognition, iris recognition and fingerprint recognition. Relations between activations for a specific spatial location are very distinctive between objects in an image. Furthermore, important activations can be extracted from the Gabor space in order to create a sparse object representation.[14]The Gabor filter is basically a Gaussian (with variances sx and sy along x and y-axes respectively) modulated by a complex sinusoid (with centre frequencies U and V along x and y-axes respectively) described by the following equation.

Texture Database:Brodatz: Brodatz textures[16] are commonly used as test data for generic texture interpretation research. Each Brodatz sample is assumed to certain only one class. Six Brodatz textures are selected for classification (Fig. 3). Each Brodatz textures are 640 x 640 pixels. These Brodatz textures include regular textures, i.e., D006,D021, D095, and nonregular textures, i.e., D029,D036,D087. [5]In the experiments, each texture image was partitioned into twenty five non over lapping sub-images with the size of 128x128 pixels. For the experiments using the textures and histogram equalized textures, each 128 x128 sub-image was down sampled to the size of 64x 64 pixels by taking the average between four adjacent pixels. To avoid the boundary problem when the sub-images were rotated, for each 128x128 sub-image, instead of down sampling, only the center64x64pixels were used in the experiments. Therefore, in all the experiments, twenty five 64x64texture samples were generated for each texture class.

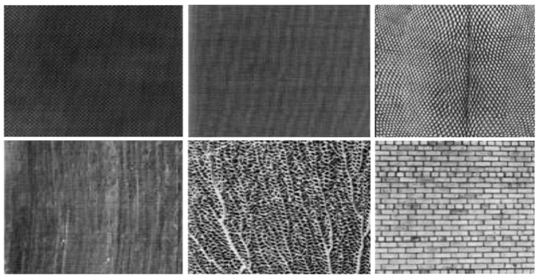

Fig 3: Six Brodatz image textures. First row from left to right: D006 (wire), D021 (french canvas), D036 (lizard cloth); second row from left to right: D079 (oriented grass-fiber cloth), D087 (fossilized sea fan), D095 (brick wall).

Outex: The Outex database contains textural images captured from a wide variety of real materials. The texture images in the Outex database are presented as test suites. Outex provides a large collection of textures and ready-made test suites for different types of texture analysis problems [12]. It is experimentally shown that our methods using combining both DLBP and Gabor filter responses can achieve high accuracy, as compared with LBP. We selected the six textures shown in (Fig. 4). The test suites comprise of different textures and are used for evaluating algorithms for various types of texture analysis. In the experiment, we made use of the Outex test suites Outex 001(pasta), 002(canvas), 004(tile), 010 (barleyrice), 011(chips), 013(seeds) for classification.

Fig 4: Six Outex image textures. First row from left to right: 001(pasta),002(canvas), 004(tile); second row from left to right: 010 (barleyrice), 011(chips), 013(seeds).

A support vector machine (SVM) is a concept in statistics and computer science for a set of related supervised learning methods that analyze data and recognize patterns, used for classification and regression analysis.The standard SVM takes a set of input data and predicts, for each given input, which of two possible classes forms the input, making the SVM a non-probabilistic binary linear classifier. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other. An SVM model is a representation of the examples as points in space, Fig 5: shows example of a SVM separating Hyper planes, H3 (green) doesn't separate the two classes. H1 (blue) does, with a small margin and H2 (red) with the maximum margin mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on.

We are given some training data D, a set of n points of the form

$$D = \{(x_i, y_i) \mid x_i \in R^p, y_i \{-1,1\}\}_{i=1}^{n}$$

Where the y_i is either 1 or -1, indicating the class to which the point X_i belongs. Each X_i is a p-dimensional real vector. We want to find the maximum-margin hyperplane that divides the points having $y_i = 1$ from those having $y_i = -1$. Any hyperplane can be written as the set of points x satisfying Maximum-margin hyperplane and margins for an SVM trained with samples from two classes. Samples on the margin are called the support vectors.

$w \bullet x - b = 0$. Where • denotes the dot product and w-the normal vector to the hyperplane. The parameter $\frac{b}{||w||}$ determines the offset of the hyperplane from the origin along the normal vector w. We want to choose the w and b to maximize the margin, or distance between the parallel hyperplanes that are as far apart as possible while still separating the data. These hyperplanes can be described by the equations.

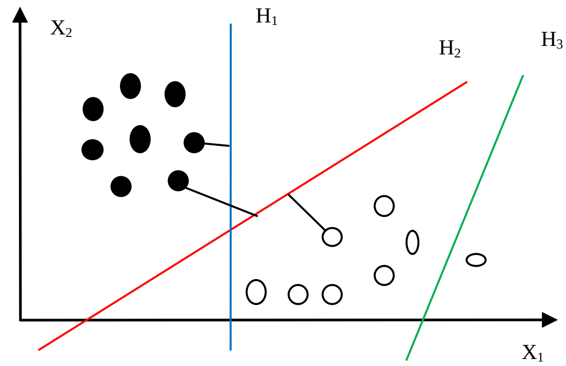

Fig 5: example of a SVM separating Hyper planes, H3 (green) doesn't separate the two classes. H1 (blue) does, with a small margin and H2 (red) with the maximum margin.

$w \bullet x - b = 0$ and $w \bullet x - b = -1$, if the training data are linearly separable, we can select the two hyperplanes of the margin in a way that there are no points between them and then try to maximize their distance. By using geometry, we find the distance between these two hyperplanes is $\frac{2}{||w||}$ so we want to minimize ||w||. As we also have to prevent data points from falling into the margin, we add the following constraint: for each i either

$w \bullet x - b \geq 1$ for x_i of the first class

or

$w \bullet x - b \leq -1$ for x_i of the second.

This can be rewritten as:

$y_i (w \bullet x - b) \geq 1$, for all $1 \leq i \leq n$.

We can put this together to get the optimization problem:

Minimize (in w,b)

||w||

Subject to (for any i=1,...,n)

$y_i (w \bullet x - b) \geq 1$

4. EXPERIMENTAL RESULTS

Each texture image has the size of 640×640 pixels and represents a texture class. Fig 6: shows the DLBP and Gabor Filter Texture classification with six Class. Each texture image was partitioned into 25 non-overlapping sub-images with the size of 128×128 pixels. By using the center 64 x 64 pixels Twenty-five 64×64 texture samples were generated for each texture class. For training used 13 samples for each texture class. 12 samples were used for validation. The image data and textures may be artificial or natural, possibly obtained in a real world application. An important part of the selection of image data is the availability and quality of the ground truth associated with the images. Each image indeed represents the texture category it is supposed to represent according to the ground truth. So called Brodatz textures were used to evaluate the performance of the proposed techniques for texture feature extraction and classification.

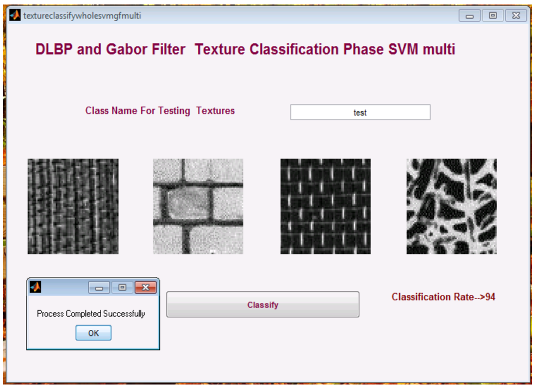

Fig 6: DLBP and Gabor Filter Texture classification.

The image data are often limited in terms of the number of original source images available, so we portioning of the image data into sub images to increase the amount of data. The method is outlined in a block diagram shown in Fig.7. The (sub) images may have different gray scale properties. Some of the (sub) images are preprocessing to have uniform gray scale distribution or equal first and second order statistics, by histogram equalization. In order to estimate the performance for classification the texture, portioning of the (sub) images data into training and testing sets should be independent the data is randomly divided into separate training and testing sets. The good features should not be wasted with poor classifier design, so we use the classification algorithm procedure to perform's the texture classification. The feature values between training and testing sets provides the basis for the quantitative performance analysis. In the case of class assignments the items in the testing set are classified, and the proportion of correctly classification accuracy. The input data will be transformed into a reduced representation set of features. A DLBP approach is used for classification of textures. The inputs to the system are the digitized images from one of the m texture classes. The images are separated into test and training set [9].

Step1: For Classification Input, take the image from the training or testing image.

Step2: Initilize the pattern histogram of size.

Step3: Read the testing for each center pixel tc.

Step4: Compute each pixel of t_C LBP $_{m,R}$ (1).

Step5: update the histogram according to the pattern

Step6: Sort the histogram in descending order.
Step7: Fetch the dominating patterns as feature vector using $K_{80\%}$
Step8: DLBP feature vector extract.

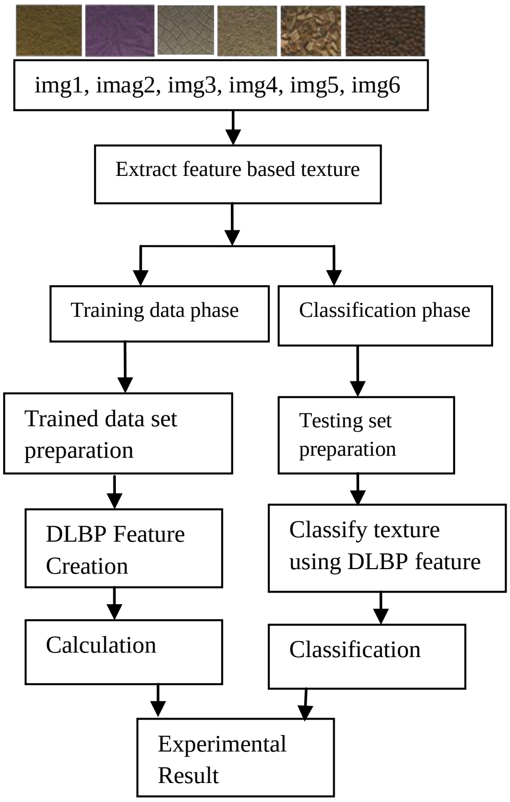

Fig 7: Block diagram of a DLBP classification.

In CDLBP (Color Dominant Local Binary Pattern) results clearly show that color and texture have complementary roles. Color histograms are more powerful than texture operators based on gray scale information in stable illumination conditions, but fail when illumination conditions change. At the same time, texture features - especially the multiresolution LBP distributions - provide fairly robust performance irrespective of illumination. The joint color texture features are not the best ones in any of the experiments. In all cases, either color or texture alone gave better accuracy.Fig.8 shows the CDLBP Texture classification. Color histograms contain very discriminative information about the color dimension of color texture. It is sur-prising that their use has not been considered in many color texture papers, it show that histograms of Local Binary Patterns provide highly discriminative pattern-related information. Fig.9 shows the RGB histograms features for Outex color texture. First, each 8-bit color channel was quantised into 16 and 32 levels by dividing the values on each color channel by 16 and 8, respectively. Three-dimensional histograms with 163 and 323 entries were created. This method is quantization method as raw quantization. Second, the quantization method presented in [18] was used in obtaining three-dimensional and one-dimensional color distributions.

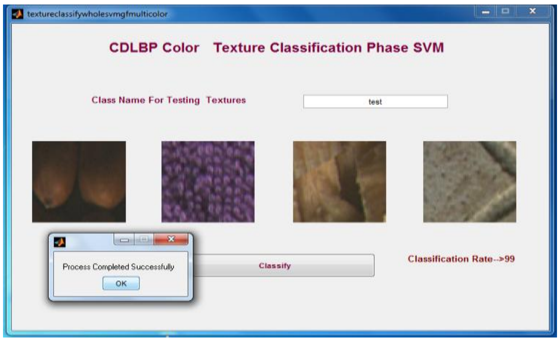

Fig 8:CDLBP Texture classification

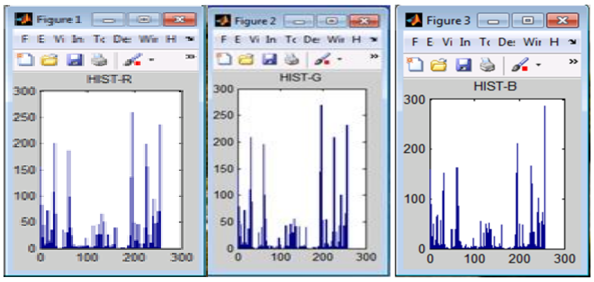

Fig 9: RGB histograms features for Outex color texture.

The resulting histograms are later denoted by RGB 16^3, 32^3 and 256 * 3. The feature based textures extract local regions of interest that is features from the images and identify the corresponding features in each image of the subsequent features of such algorithms have initially been developed for tracking a small number of salient features in long image sequences. The tracking process can be of subtasks. Feature extraction and Feature tracking. One can extract features only in the first image and search for the same features in the subsequent images. The dynamic feature extraction Scheme is advocated by the feature tracker [9] [10]. Transforming the input data into the reduced set of features is called feature extraction. Feature extraction is when an input data to an algorithm is too large to be processed and it is suspected to be notoriously redundant that is it consists of much data not much information. If the features extracted are carefully chosen it is expected that the features set will extract that the relevant information from the input data in order to perform the desired task using this reduced representation instead of the full size input. Feature extraction involves simplifying the amount of resources required to describe a large set of data accurately. Analysis with a large number of variables generally requires a large amount of memory and computation power or a classification phase, which over fits the training sample and generalizes poorly to new samples.

5. DISCUSSIONS AND CONCLUSIONS

The DLBP approaches were able to represent the dominant patterns in the texture images. It retains the rotation invariant and histogram equalization invariant properties of the conventional LBP approach. It is computationally efficient. Regarding the selection of the first 80% most frequently appeared patterns as features, after the pattern frequency histogram is constructed and sorted for the DLBP for color texture image. It gives good classification results, excellent classification performance are demonstrated. By comparing both LBP and DLBP gives more matching. The experiment results show the proposed method outperforms the other image features in terms of the classification rates in various image conditions.

In future the application can extended for various color texture images with DLBP and CDLBP features.

6. REFERENCES

[1] S.Liao, Max W. K. Law, and Albert C. S. Chung, "Dominant local binary patterns for texture classification," IEEE Trans. Pattern Anal. Mach. Intell., vol. 18, no. 5, pp. 1107–1118, May. 2009.

[2] H. Arof and F. Deravi, "Circular neighborhood and 1-D DFT features for texture classification and segmentation," Proc. IEE Vision, Image and Signal Process., vol. 145, no. 3, pp. 167–172, Jun. 1998.

[3] A. C. Bovik, M. Clark, and W. S. Geisler, "Multichannel texture analysis using localized spatial filters," IEEE Trans. Pattern Anal. Mach. Intell., vol. 12, no. 1, pp. 55–73, Jan. 1990.

[4] R. Chellappa and S. Chatterjee, "Classification of textures using Gaussian Markov random fields," IEEE Trans. Acoust., Speech, Signal Process., vol. ASSP-33, no. 4, pp. 959–963, Apr. 1985.

[5] H.Deng and D. A. Clausi, "Gaussian MRF rotation-invariant features for image classification," IEEE Trans. Pattern Anal. Mach. Intell., vol.26, no. 7, pp. 951–955, Jul. 2004.

[6] G. M. Haley and B. S. Manjunath, "Rotation-invariant texture classification using a complete space-frequency model," IEEE Trans. Image Process., vol. 8, no. 2, pp. 255–269, Apr. 1999.

[7] X. Huang, S. Z. Li, and Y. Wang, "Shape localization based on statistical method using extended local binary pattern," in Proc. 3rd Int. Conf. Image and Graphics, 2004, pp. 184–187.

[8] L. M. Kaplan, "Extended fractal analysis for texture classification and segmentation," IEEE Trans. Image Process., vol. 8, pp. 1572–1585, 1999.

[9] R. L. Kashyap and A. Khotanzad, "A model-based method for rotation invariant texture classification," IEEE Trans. Pattern Anal. Mach. Intell., vol. 8, no. 7, pp. 472–481, Jul. 1986.

[10] S. Li and J. Shawe-Taylor, "Comparison and fusion of multiresolution features for texture classification," Pattern Recognit. Lett,vol. 26, pp. 633–638, 2005.

[11] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," IEEE Trans. Pattern Anal. Mach. Intell., vol. 11, no. 7, pp. 674–693, Jul. 1989.

[12]T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," IEEE Trans. Pattern Anal. Mach. Intell., vol. 24, no. 7, pp. 971–987, Jul. 2002.

[13] R. Porter and N. Canagarajah, "Robust rotation-invariant texture classification: Wavelet, Gabor filter and GMRF based schemes," IEE Proc. Vis., Image, Signal Process., vol. 144, no. 3, pp. 180–188, 1997.

[14] H.Yao, B. Li, and W. Cao, "Remote sensing imagery retrieval based-on Gabor texture feature classification," in Proc. 7th Int. Conf. Signal Processing, 2004, vol. 1, pp. 733–736.

[15] J. Zhang, T. N. Tan, and L. Ma, "Invariant texture segmentation via circular Gabor filters," in Proc. 16th IEEE Int. Conf. Pattern Recognition, 2002, pp. 901–904.

[16] P. Brodatz, Textures: A Photographic Album for Artists and Designers. New York: Dover, Jun.1996.

[17] T. Ojala, T. Maenpaa, M. Pietikainen, J. Viertola, J. Kyllonen, and S. Huovinen, "Outex-A New Framework for Empirical Evaluation of Texture Analysis Algorithms",Proc. 16th Int'l Conf. Pattern Recognition, 2002.

[18]T. Ojala, , M.Pietikainen and D. Harwood, "A ComparativeStudy of Texture Measures with Classification Based on Fea-ture Distributions," Pattern Recognition, vol. 29, pp. 51- 59,1996.