

AN EFFECTIVE GLCM AND BINARY PATTERN SCHEMES BASED CLASSIFICATION FOR ROTATION INVARIANT FABRIC TEXTURES

R. Obula Konda Reddy¹, Dr. B. Eswara Reddy², Dr. E. Keshava Reddy³

Research Scholar, JNTUA College of Engineering, India¹, HOD, Department of CSE, JNTUA College of Engineering, India², Department of Mathematics, JNTUA College of Engineering, India³

reddyobulakonda@gmail.com¹, eswararedd@gmail.com², reddyekeshava@gmail.com³

ABSTRACT

Textures are one of the basic features in visual searching, computational vision and also a general property of any surface having ambiguity. This paper presents a novel texture classification system which has a high tolerance against illumination variation. A Gray Level Co-occurrence Matrix (GLCM) and binary pattern based automated similarity identification and defect detection model is presented. Different features are calculated from both GLCM and binary patterns (LBP, LLBP, and SLBP). Then a new rotation-invariant, scale invariant steerable decomposition filter is applied to filter the four orientation sub bands of the image. The experimental results are evaluated and a comparative analysis has been performed for the four different feature types. Finally, the texture is classified by different classifiers (PNN, KNN and SVM) and the classification performance of each classifier is compared. The experimental results have shown that the proposed method produces more accuracy and better classification rate over other methods.

Index Terms— Feature extraction, KNN, LBP, LLBP, SLBP, Steerable filter decomposition

INTRODUCTION

Texture is also a main visual feature that refers to natural and fundamental surface properties of an object and their relationship with the surrounding atmosphere. It can be seen practically anywhere. The texture can be regarded as the visual look of a surface or material. Typically, textures and the analysis techniques associated with them are divided into two major categories with dissimilar computational approaches: the stochastic and the structural methods. With a very regular appearance structural textures are frequently man-made. In structural texture analysis, the properties and the appearance of the textures are described with a dissimilar policy that specifies what kind of primitive elements there are in the surface and how they are located [14]. Texture classification is a primary issue in computer vision and image processing, playing a very significant role in a wide range of applications that include medical image analysis, remote sensing, object recognition, document analysis, environment modeling, content-based image retrieval and many more. Local binary pattern (LBP) is one of the most used texture descriptors in image analysis [7].

Textures provide essential information for many image classification tasks. Much research has been done on texture classification during the last three decades; most traditional approaches included gray level co-occurrence matrices (GLCM), second-order statistic method, Gauss–Markov random field and local linear transform, which are restricted to the analysis of spatial relations between neighboring pixels in a small image region. Recently, effective measurement of local brightness variations and local texture softness by using features such as BDIP (block difference of inverse probabilities) and BVLC (block variation of local correlation coefficients) have been proposed. As a consequence, their performance is the best for the class of so called micro-textures [6] [12]. Texture classification plays an

important role in engineering fields and scientific researches. It can be used in image database retrieval, industrial, agricultural and biomedical applications as well. The training phase and the testing phase/recognition phase are the two different process involved in texture image classification. In the training phase a set of known texture images are trained by the feature extraction method and stored in the library or database. In the recognition phase the unknown sample image is tested by using the same feature extraction method and the values are compared with the already stored features in the database [11].

Texture is characterized not only by the gray value at a given pixel, but also by the gray value pattern in a neighborhood near the pixel. Generally, the term texture refers to repetition of texture primitives or fundamental texture elements called texels. A texel contains a number of image pixels, whose placement may be periodic, quasi-periodic, or random. The category of the texture determined by the repetitiveness of the texels and the texture analysis approach is decided [8]. A texture image is principally a task of the subsequent variables: the texture surface, its albedo, the camera, the illumination and its viewing arrangement. It is a mixture of all these factors which makes the texture classification a difficult task. Due to the increasing demand of such applications, texture classification has received substantial attention over the last several decades and several novel methods have been proposed [15]. One of the most important problems is that the textures in the real world are often dissimilar due to variations in orientation, size or other visual appearance. Due to irregular illumination or large within-class inconsistency the gray-scale invariance is frequently significant. Moreover, most proposed texture measures have a high degree of computational complexity.

In various applications, to deal with the semantic gap trouble, texture features are employed. For example, it is used to explain topographical surfaces in satellite images and organ's tissues in the medical imaging field. Thus, the majority of the research in the area of texture analysis is devoted to developing the inequitable capability of the features extracted from the image. Texture analysis is typically a very lengthy practice [5]. A neural network as a classifier is used in many texture identification systems. Neural networks have been paying attention to researchers in pattern recognition because of its power to learn from training datasets. An example would be the back-propagation for the adaptive route selection policy in mobile ad hoc networks. PNN is another neural network that has been used in several applications. The applications are remote sensing image segmentation and surface defect identification. PNN has verified to be efficient more than conventional back-propagation based networks and has been accepted as an substitute in realtime categorization problems [10].

In order to devise an efficient algorithm for texture classification it is necessary to find a set of texture features with good discerning power. Most of the textural features are usually obtained from the application of a local operator, statistical analysis or dimension in a transformed domain. In general, the features are predicted from co-occurrence matrices, Law's texture energy measures, Fourier transform domain, Markov random field models, local linear transforms etc [16]. Local Binary Patterns (LBP), a non-parametric technique summarizing the local structures of an image efficiently has received growing interest for facial representation recently. LBP was initially proposed for texture description and has been broadly exploited in numerous applications. The most important properties of LBP features are tolerance against the monotonic illumination changes and computational simplicity as well [20]. In recent years, LBP features have been extensively exploited for facial image analysis, together with face detection, face recognition; facial expression analysis, gender/age categorization and some other applications. In the meantime, different variations of the original LBP have been proposed for an improved performance.

This work addresses the above mentioned issues. In this paper, a collective methodology is proposed for both scale and rotation invariant classification. The rest of the paper is organized as follows: a literature review of the existing texture classification methods is given in section 2.

2. RELATED WORKS

Alireza Tavakoli Targhi et al. [19] have proposed a technique for texture classification with minimal training images. They have classified texture from a single image under unknown lighting conditions. The current and successful approach to this task is to treat it as a statistical learning problem and learn a classifier from a set of training images, but this required a not only a sufficient number but also a variety of training images. They showed that the number of training images required can be drastically reduced (to as few as three) by synthesizing additional training data using photometric stereo. They demonstrated the method on the PhoTex and ALOT texture databases. Despite the limitations of photometric stereo, the resulting classification performance surpassed the state of the art results.

Dattatraya S. Bormane et al. [1] have developed a scale invariant texture classification method based on Fuzzy logic. It was applied for the classification of texture images. Two types of texture features were extracted; one using Discrete Wavelet Transform (DWT) and other using Co-occurrence matrix. Co-occurrence features were obtained using DWT coefficients. Two features were obtained from each sub-band of DWT coefficients up to the fifth level of decomposition and eight features were extracted from the co-occurrence matrix of the whole image and each sub-band of the first level of DWT decomposition. The fuzzy classification was achieved in two steps, fuzzification and rule generation. The performance was measured in terms of Success Rate. This study showed that the proposed method offered excellent scale invariant texture classification Success Rate. Also, wavelet features like standard deviation and combination of energy along with some proposed hybrid feature sets outperformed the other feature sets. This success rate was comparatively high when compared with results published earlier.

Jing Yi Tou et al. [17] have analyzed recent trends in texture classification in which they have aimed to compile the recent trends on the usage of feature extraction and classification methods used in the research of texture classification as well as the texture datasets used for the experiments. The study showed that the signal processing methods, such as Gabor filters and wavelets were gaining popularity, but old methods such as GLCM were still used but were being improved with new calculations or combined with other methods. For the classifiers, nearest neighbor algorithms were still fairly popular despite being simple and SVM has become a major classifier used in texture classification.

3. MAIN CONTRIBUTIONS

First, this work enhances the recent technique on texture classification for better dealing with rotation invariant fabric images. Binary pattern features are extracted and a steering filter is applied to them at different orientations. GLCM features are then extracted and as the ultimate step, two types of classifications such as, similarity and defect classifications are performed on the input fabric texture images. Finally, our experiments with three different patterned fabric images (star, box and dot) and with the three different classifiers provide a better classification rate.

The outline of this paper is as follows. The proposed texture classification algorithm and a detailed review about the various features and classifiers considered in this work are described in Section 4. Section 5 provides a detailed mathematical model of the proposed texture classification. Section 6 illustrates the experimental results and evaluates the performance of the proposed system. Then the conclusion of this work is summarized in Section 7. Finally, the paper ends with the references.

4. PROPOSED METHODOLOGY

The proposed system for texture classification is built based on GLCM, LBP, LLBP, SLBP and by applying KNN, SVM and PNN as classification algorithms. The intensity of the image is first normalized and after that, binary patterned images are constructed and a steering filter is applied at different rotational angles {0, 45,-45,180}. After the extraction of the GLCM features, similarity and defect classification is performed using classifiers like SVM, KNN and PNN. For defect classification, the following steps are performed. First, the defect free image of different patterns were used for training. Then the feature vectors of $k \times l$ dimensions are acquired for these images, where “k” is the number of training images and “l” is the dimension of the features. Here, different defective images like broken end,

hole, thick bar, thin bar and netting multiple [2] were used for testing/classification. Similarity based classification is done by using the same steps like defect classification.

The theoretical background of all the approaches employed and the overall processing steps are discussed in this section.

4.1. Intensity Normalization

Many papers attempt to get some face feature which is insensitive to the variation in illumination. With the feature, the varying illumination on the image cannot manipulate the classification result. In other words, we can eradicate the illumination factor from the image. The best way to do this is to separate the intensity information from the identity information.

The image intensity is normalized by applying a technique called histogram equalization which is described as follows.

4.1.1. Histogram Equalization

Histogram equalization (HE) is a classic technique. It is generally used to make an image with a uniform histogram, which is considered to produce an optimal global contrast in the image. However, HE may make an image under irregular illumination to be more irregular. S.M. Pizer and E.P. Amburn (Pizer & Amburn, 1987) proposed an adaptive histogram equalization (AHE). It calculates the histogram of a local image region centered at a given pixel to determine the mapped value for that pixel; this can accomplish a local contrast enhancement.

4.2. Feature Extraction

Feature Extraction is a major process in all classification and recognition applications. For texture classification several feature extraction methods are available. Few of them are Binary patterns, GLCM, GLWM, TGF, DBWP etc. Our proposed texture classification considers two feature types, such as GLCM and binary patterns which are detailed in the following sub sections.

4.2.1. Binary Patterns

Binary pattern is an image formed by a formula that consists of binary operations and results in a 32-bit integer number. These patterns are closely united to the 32-bit RGB color system. Any integer numbers can use these patterns. The different binary patterns considered in this work are Local Binary Pattern (LBP), Simplified Local Binary Pattern (SLBP) and Local Line Binary Pattern (LLBP).

4.2.1.1 LBP

The local binary pattern (LBP) texture analysis operator is defined as a gray-scale invariant texture measure derived from a common definition of texture in a local region. It is an effortless but efficient texture operator which labels the pixels of an image by the process of thresholding the neighborhood of each pixel and results in a binary number. It became an admired technique due to its discriminative power and computational simplicity [4]. Ever since the LBP was, by definition, invariant to monotonic alterations in gray scale, it was incremented by an orthogonal assessment of local contrast. Fig 1 shows how the contrast measure (C) is derived. The gray level average below the center pixel is subtracted from that of the gray levels that are above the center pixel. The LBP and local contrast measures are used as two dimensional distribution features.

i_0	i_1	i_2
i_3	i_4	i_5
i_6	i_7	i_8

(a)

1	2	4
128	0	8
64	32	16

(b)

Fig 1: (a) LBP operator binary sequence (b) weighted thresholds

4.2.1.2. SLBP

Simplified local binary pattern (SLBP) was proposed by Qian Tao and Reymaond Veldhuis for illumination normalization. This is calculated by allotting equal weights to each of the 8 near regions. SLBP shows that the processed image becomes more robust to illumination changes. It has the following merits: the simplified one is not directional-sensitive and the coding number is largely reduced from 256 to 9 patterns.

4.2.1.3. LLBP

Local line binary pattern (LLBP) is derived from the local binary pattern (LBP) and it reveals the local spatial structure of an image by the process of thresholding. The texture presentation denoted as a decimal number is initiated from the local window with a binary weight. LLBP has a very low computational cost. The fundamental idea of both LLBP and LBP are same but the main differences are as follows:

- 1) The LLBP region is a straight line with N length pixel, where as the LBP region is square in shape.
- 2) Binary weight distribution is started from left and right adjacent pixels of the center pixel, whereas in LLBP it is started from the top and bottom adjacent pixels of the center pixel. .

The LLBP algorithm gets the line binary code along with horizontal and vertical directions separately and also its magnitude, which differentiates image intensity changes such as corners and edges as well.

4.3. GLCM

Gray-level co-occurrence matrix (GLCM) is a statistical way of investigative texture that considers the spatial relationship of pixels. It is also recognized as the gray-level spatial dependence matrix. After creating a GLCM by estimating frequent pairs of pixels with precise values in an image, statistical measures are extracted. This is an Ng dimension square matrix, where Ng is the number of gray levels in the image. Element [i, j] of the matrix is generated by counting the number of times a pixel with value i occur as a neighbor to a pixel with value j and then dividing the whole matrix by the total number of such relationships made [3]. Thus, each entry is measured as the probability that a pixel with value i will be found near a pixel of value j. The GLCM can be defined as:

$$P^0(x, y) = \Pr(I(p_1) = x \wedge I(p_2) = y \wedge \|p_1 - p_2\| = d) \quad (1)$$

Where P is the probability, p1 and p2 are the positions in the gray scale image I.

4.4. Steerable filter

In vision and image processing tasks, oriented filters are used for image data compression, motion analysis and texture analysis. By applying many versions of the same filter, the response of a filter gets much orientation is found in which each of them is made different from the other by doing some rotation in small angles. Steerable filters are spatial oriented filters that can be expressed using linear combinations of a permanent set of basis filters. The filter is said to be shiftable in position if the transformation is a translation. And if the transformation is a rotation, then the filter is said to be steerable in orientation. The basis filters are normally called steerable basis filters. Given a set of steerable basis filters, we can apply them to an image and since convolution is linear, we can interpolate accurately from the responses of the basis filters the output of a filter tuned to any orientation we desire [9].

A steerable filter is a kind of filter which can filter arbitrary orientation which is produced as a linear combination of the benefits of the “basis filter”. The edge at different orientations in an image can

be located by splitting the image into orientation sub-bands. Steerable filters allow one to adjust any orientation i.e., to "steer" a filter and to analytically determine the filter output as a function of orientation.

The Steering Constraint is defined as:

$$G_{\theta}(m, n) = \sum_{l=1}^N b_l(\theta) A_l(m, n) \quad (2)$$

Here $b_l(\theta)$, θ controls the filter orientations and b_l is the interpolation function based on the arbitrary orientation. The basis filter $A_l(m, n)$ is a rotated version of impulse response θ .

$$(m, n) = \sum_{l=1}^N b_l(\theta) (I(m, n)) A_l(m, n) \quad (3)$$

Generally speaking, the texture image can be seen as a set of basic cyclic primitives characterized by their spatial homogeneity. By applying statistical measures, this information is extracted and used to capture the related image content as feature vectors. More precisely, we use the mean μ and standard deviation σ of the filtered images $S_j(x, y)$. And we use contrast, correlation, angular second moment, entropy, difference entropy by considering the presence of homogeneous regions in the texture images. Here $j=1,2,3,4$ represent horizontal orientation, rotation of 45° , vertical orientation, and rotation of -45° , respectively. Given an image $I(x, y)$, its steerable filter decomposition is:

$$S_j(x, y) = \sum_{x_l} \sum_{y_l} I(x_l, y_l) C_j(x - x_l, y - y_l) \quad (4)$$

Where C_j denotes the directional band pass filter at orientations, $j=1, 2, 3, 4$.

4.5. Classifiers

Three different types of classifiers (KNN, PNN and SVM) are employed in this work for similarity and defect classification of textures. All of them are described in brief in the following three subsections.

4. 5.1. KNN Classifier

In pattern recognition, the k-nearest neighbor (K-NN) algorithm is a technique for classifying objects based on closest training sample in the feature space. K-NN is a type of instance-based learning having a locally approximated function and all the computations are postponed until classification. In K-NN, an object is classified based on the majority vote of its neighbors with the object being assigned to the most common class amongst its k nearest neighbors (k is a positive integer, typically small). If the value of k is 1, then the object is just assigned to its nearest neighbor class [3]. The neighbors are taken from a set of correct classification known objects. This is considered as the training set for the algorithm, even if no explicit training step is required.

4. 5.2. PNN Classifier

Equations within the structure are selected by the Probabilistic Neural Networks using Statistical methods. The PNN is used to categorize the patterns based on learning from instance. It is based on the algorithm of "The Bayes Strategy for Pattern Classification". Dissimilar rules determine pattern statistics from the training examples. A Probabilistic Neural Network is designed as one containing one input layer, and two hidden layers. When an input is provided, the first layer computes distances from the input vector to the training input vectors and generates a vector whose elements specify the closeness of the input with a training input [13]. The second layer totals these contributions for each class of inputs to produce a vector of probabilities as its net output. Finally, a maximum of these probabilities is picked by the entire transfer function on the output of the second layer and gives an output of a 1 for that class and a 0 for the

other classes. The main benefit of PNN is that they are very flexible and with almost no retraining, new information can be added immediately.

4.5.3. SVM Classifier

In the environment of two-class classification problems, SVM is a learning system that uses an optimal hyperplane to classify sets of feature vectors into two classes. During the training phase, the hyperplane which has the maximum margin of separation between the two classes is designed as an optimal one [18]. Once the training phase is completed, the test phase begins in which a feature vector not used for training is then classified which fits in to either one of the two classes based on its position with respect to the optimal hyperplane. To compare with a number of classes k larger than two, the one against-one approach is used, i.e., $k(k - 1)/2$ SVM classifiers being trained on feature vectors from two class memberships only are built, then a test feature vector is linked with the class membership to which it is more frequently associated by the different SVM classifiers.

4.6. Proposed block diagram

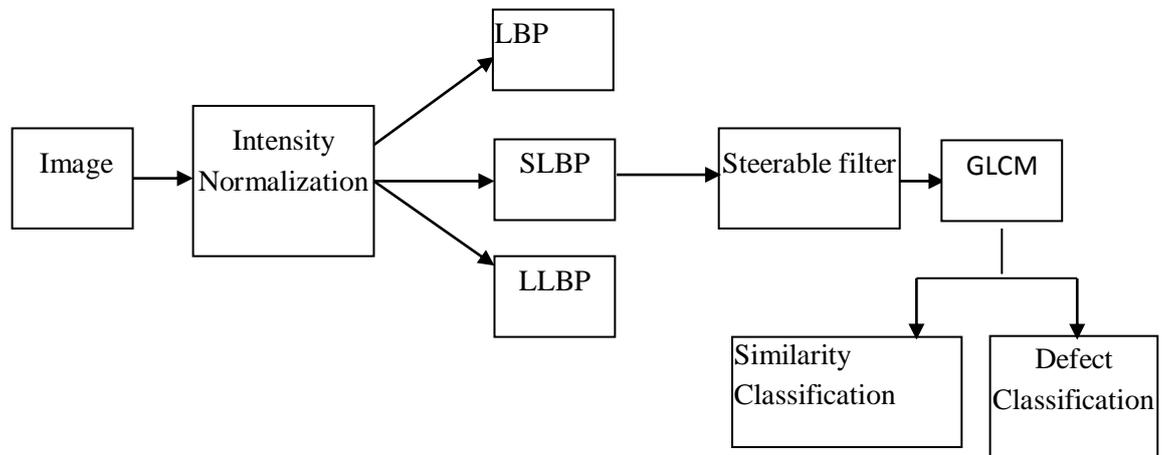
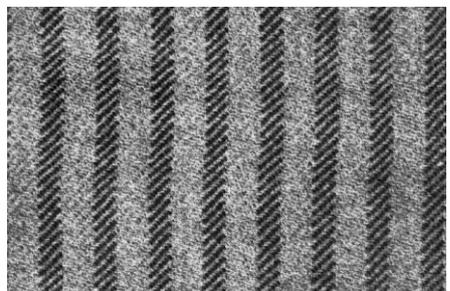
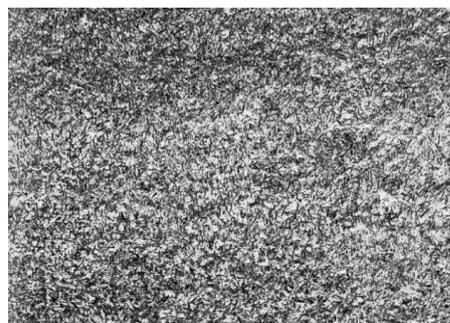


Fig 2: Block diagram of the proposed system

5. TEXTURE CLASSIFICATION ON FABRIC IMAGES:

Some of the sample images used in the experiment are provided in fig 3.



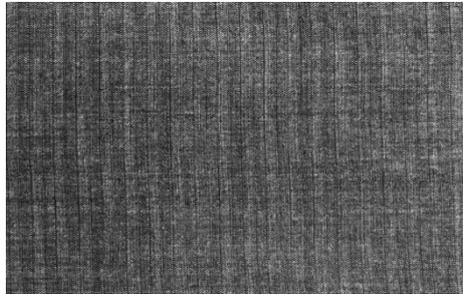


Fig 3: Sample images

5.1. Binary Patterns:

Consider an image $I(x, y)$ of size $m \times n$ and let g_l denote the gray level of an arbitrary pixel (x, y) , i.e. $g_l = I(x, y)$. Moreover, let g_s denote the gray value of a sampling point in an evenly spaced circular neighborhood of sampling points and radius R around point (x, y) :

$$g_s = I(x_s, y_s), s = 0 \dots S - 1 \quad (5)$$

$$x_s = x + R \cos(2\pi s / S) \quad (6)$$

$$y_s = y - R \sin(2\pi s / S) \quad (7)$$

Assuming that the local texture of the image $I(x, y)$ is characterized by the joint distribution of gray values of $S+1$ ($S > 0$) pixels:

$$T = t(g_l, g_0, g_1, \dots, g_{S-1}) \quad (8)$$

Without loss of information, the center pixel value can be subtracted from the neighborhood:

$$T = t(g_l, g_0, g_1, g_1 - g_l, \dots, g_{S-1} - g_l) \quad (9)$$

In the next step the joint distribution is approximated by assuming the center pixel to be statistically independent of the differences, which allows for factorization of the distribution:

$$T \approx (g_l) t(g_0 - g_l, g_1 - g_l, \dots, g_{S-1} - g_l) \quad (10)$$

Now, the first factor $t(g_l)$ is the intensity distribution over $I(x, y)$. The joint distribution of differences can be used to model the local texture. Only the signs of the differences are considered: Where $f(z)$ is the thresholding (step) function.

$$(g_0 - g_l, g_1 - g_l, \dots, g_{S-1} - g_l) \quad (11)$$

$$f(z) = \begin{cases} 1 & z \geq 0 \\ 0 & z < 0 \end{cases} \quad (12)$$

From this joint distribution, the generic local binary pattern operator is derived. As in the case of a basic LBP, it is gained by summing the thresholded differences weighted by powers of two. The LBP_{P,R} operator is defined as

$$LBP_{P,R}(x_l, y_l) = \sum_{s=0}^{s-1} f(g_s - g_l) \cdot 2^s \quad (13)$$

Similarly for SLBP and LLBP the mathematical representations are given in equations

$$SLBP(x_d, y_d) = \sum_{m=0}^7 S(i_m - i_c) \cdot 1 \quad (14)$$

$$LLBP_h(N, D) = \sum_{n=1}^{d-1} S(h_m - h_d) \cdot 2^{(d-m-1)} + \sum_{m=d+1}^N [S(h_m] - h_d) \bullet 2^{(m-d-1)} \quad (15)$$

$$LLBP_v(N, D) = \sum_{m=1}^{d-1} S(v_m - v) \cdot 2^{(d-m-1)} + \sum_{m=d+1}^N [S(v_m] - v_d) \cdot 2^{(m-d-1)} \quad (16)$$

$$LLBP_m = \sqrt{LLBP_h^2 + LLBP_v^2} \quad (17)$$

The energy distribution $E_j(x, y)$ of the filtered images $S_j(x, y)$ is defined as

$$E_j = \sum_x \sum_y |S_j(x, y)| \quad (18)$$

The mean (μ_j) and standard deviation (σ_j) are found as follows:

$$\mu_j = \frac{1}{MN} E_j(x, y) \quad (19)$$

$$\sigma_j = \sqrt{\frac{1}{MN} \sum_x \sum_y [(S]j(x, y) - \mu_j)^2} \quad (20)$$

5.2. GLCM

Given an image, composed of pixels each with an intensity (a specific gray level), the GLCM is a tabulation of how often dissimilar arrangements of gray levels co-occur in an image. Texture feature calculations employ the GLCM information to provide a variation measure of the intensity at the pixel of interest. Create a GLCM square matrix of size $N \times N$ where N is the number of quantization levels. Then, the features such as contrast, correlation, angular second moment, entropy and difference entropy are calculated.

The Contrast feature is calculated as:

$$f_{c1} = \sum_{j,k=0}^{N-1} P_{jk} (j-k)^2 \quad (21)$$

The Correlation feature is calculated as:

$$f_{c2} = \sum_{j,k=0}^{N-1} P_{jk} \frac{(j-y)(k-y)}{\sigma^2} \quad (22)$$

The Entropy feature is calculated as:

$$f_e = \sum_{j,k=0}^{N-1} -\ln(p_{jk}) p_{jk} \quad (23)$$

The Angular Second Moment feature is calculated as:

$$f_w = \sum_j \sum_k \{p(j,k)\}^2 \quad (24)$$

The difference entropy feature is calculated as:

$$f_{de} = - \sum_{j=0}^{N-1} p_{x-y}(j) \log\{p_{x-y}(j)\} \quad (25)$$

Where

P_{jk} = Element j, k of the normalized symmetrical GLCM

N = Number of gray levels in the image

y = GLCM mean which is calculated as follows

$$y = \sum_{j,k=0}^{N-1} j p_{jk}$$

σ^2 = intensity variation of all reference pixels which is calculated as follows

$$\sigma^2 = \sum_{j,k=0}^{N-1} p_{jk} (j-y)^2$$

So, the corresponding texture feature vectors of the original image I(x, y) are defined as:

$$TF = \left\{ \mu_1, \sigma_1, \mu_2, \sigma_2, \mu_3, \sigma_3, \mu_4, \sigma_4, f_{c11}, f_{c21}, f_{\theta 1}, f_{e1}, f_{de1}, f_{c12}, f_{c22}, f_{\theta 2}, f_{de2} \right\} \quad (26)$$

Thus, a set of feature vectors is obtained as in equation (18). For experimental analysis we use fabric images of different patterns like star, box and dot.

6. EXPERIMENTAL RESULTS

A detailed experimental result analysis is provided in this section. The proposed methodology was implemented using MATLAB 7.10 tool and was evaluated by testing the proposed scheme with different input of fabric texture images. Fig.3 shows the classification result for defect data based on binary patterns. Both binary patterns and the GLCM based classification result for defect data is exposed in fig 4. The subsequent figures show the similarity classification results for binary patterns, binary patterns and GLCM respectively.

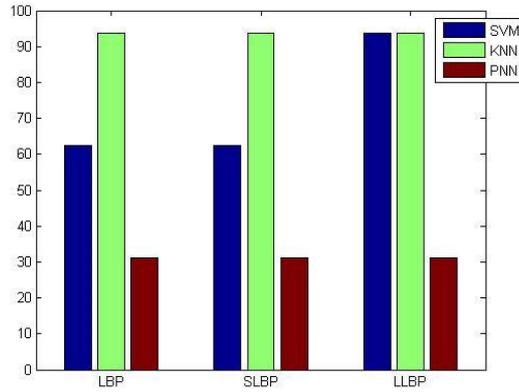


Fig 4: Defect classification result for binary patterns

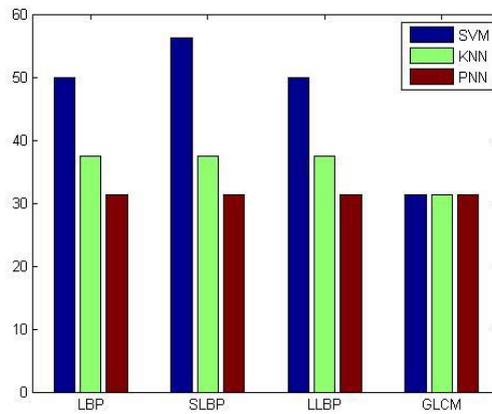


Fig 5: Defect classification result for binary patterns and GLCM

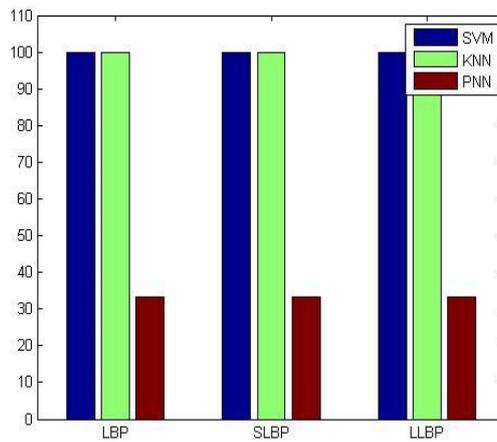


Fig 6: Similarity classification result for binary patterns

The table of results is provided in the following section. Table 1 reveals the classification rate for defect textures based on the classifiers SVM, KNN and PNN.

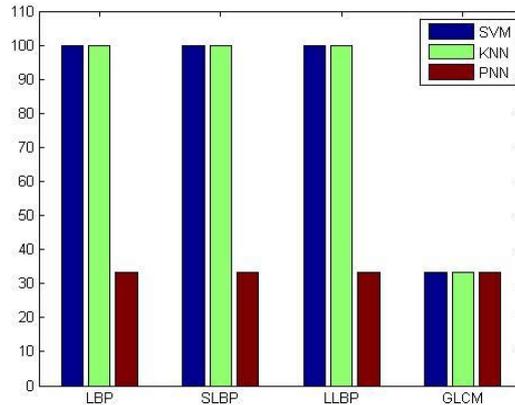


Fig 7: Similarity classification result for binary patterns and GLCM

Similarity Classification rate for textures based on the classifiers SVM, KNN and PNN is shown in table 2. From the first table it is obvious that both SVM and KNN give exactly the same classification rate for the features used. They give a 100% classification rate for both the binary patterns and for GLCM combined with binary patterns. They give a better performance than PNN for defect classification of textures. Table 2 shows that for similarity classification, SVM provides a better result for the binary pattern SLBP but a much higher result for GLCM combined with binary patterns. KNN provides a better classification rate for GLCM combined with binary patterns than the binary patterns. PNN provides the same classification rate for both binary patterns and GLCM combined binary patterns.

Table 1: Defect classification rate

Features	Classification rate		
	SVM	KNN	PNN
LBP	100	100	33.3333
SLBP	100	100	33.3333
LLBP	100	100	33.3333
GLCM	33.3333	33.3333	33.3333
LBP GLCM	100	100	33.3333

tested with the best quality fabric images and two analysis classifications and detection have been sought. From the above results it can be concluded that all the binary patterns and GLCM combined binary patterns prove to be better in measuring defect classification for both SVM and KNN classifiers. For similarity classification, both SVM and KNN provide a much higher result for GLCM combined with binary patterns than for binary patterns. It is shown that both SVM and KNN offer the highest classification rate of 100% and the highest classification rate of 33.3333% for PNN. This can be further extended in the future with the addition of different types of defect textures and also with additional training images.

REFERENCES

- [1] Dattatraya, S.Bormane, Shailendrakumar, M. Mukane,Sachin and R. Gengaje, "Rotation Invariant Texture Classification using Fuzzy Logic", International Journal of Computer Application, vol. 41,no. 1,pp. 41-44, 2012.
- [2] R.S.Sabeenian, M.E.Paramasivam and P.M.Dinesh, "Fabric defect detection in handlooms cottage silk industries using image processing Techniques", International Journal of Computer Applications, vol. 58, no. 11, pp. 21-29, 2012.
- [3] A. Suresh and K. L. Shunmuganathan, "Image Texture Classification using Gray Level Co-Occurrence Matrix Based Statistical Features", European Journal of Scientific Research, vol.75, no.4 , pp. 591-597, 2012.
- [4] C. Callins Christiyana and V.Rajaman, "Comparison of Local Binary Pattern Variants for Ultrasound Kidney Image Retrieval", International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 2, Iss. 10, pp. 224-228, October 2012.
- [5] Alceu Ferraz Costa, Gabriel Humpire-Mamani and Aigma Juci Machado Traina,"An Efficient Algorithm for Fractal Analysis of Textures", Graphics, Patterns and Images (SIBGRAPI) Conference, pp.39-46, 2012.
- [6] Manoj Kumar, Sagun Kumar sudhansu and Kasabegoudar, "Wavelet Based Texture Analysis and Classification with Linear Regration Model", International Journal of Engineering Research and Applications, vol. 2, Iss. 5, pp. 1963-1970, 2012.
- [7] Hamid Reza Eghtesad Doost, "An Efficient Method for Texture Classification with Local Binary Pattern Based on Wavelet Transformation", International Journal of Engineering Science and Technology, vol. 4, No. 12,pp. 4881-4885,2012.
- [8] E.M.Srinivasan, K.Ramar and A.Suruliandi, "Rotation Invariant Texture Classification using Fuzzy Local Texture Patterns", International Journal of Computer Science and Technology, vol. 3, Iss. 1, 2012.
- [9] Mahesh1 and M.V.Subramanyam, "Invariant Corner Detection Using Steerable Filters and Harris Algorithm", An International Journal on Signal & Image Processing, vol. 3, No. 5, pp. 111-118, 2012.
- [10] Abdul Kadir, Lukito Edi Nugroho, Adhi Susanto and Paulus Insap Santosa,"Leaf Classification Using Shape, Color, and Texture Features", International Journal of Computer Trends and Technology, pp. 225-230, 2011.
- [11] R.S.Sabeenian and V.Palanisamy, "Texture Image Classification using Multi Resolution Combined Statistical and Spatial Frequency Method", International Journal of Technology And Engineering System(IJTES), Vol. 2, No. 2, pp. 167-171, 2011.
- [12] Xiang-Yang Wang, Yong-Jian Yu, Hong-Ying Yang, "An effective image retrieval scheme using color, texture and shape features", Computer standard and interfaces, pp. 59-68,2010.
- [13] Barron, Jonathan Malik and Jitendra, "Discovering Efficiency in Coarse-To-Fine Texture Classification", June 12, 2010.

- [14] R.Suguna and P.Anandhakumar, "A Rotation Invariant Pattern Operator for Texture Characterization", International Journal of Computer Science and Network Security, vol. 10, No. 4, pp. 120-129, 2010.
- [15] S. Liao, Max W. K. Law and Albert C. S. Chung, "Dominant Local Binary Patterns for Texture Classification", IEEE Transactions on Image Processing, vol. 18, no. 5, pp. 1107-1118, May 2009.
- [16] B.V. Ramana Reddy, A. Suresh, M. Radhika Mani and V.Vijaya Kumar, "Classification of Textures Based on Features Extracted from Preprocessing Images on Random Windows", International Journal of Advanced Science and Technology, vol. 9, pp. 9-18, 2009.
- [17] Jing Yi Tou, Yong Haur Tay and Phooi Yee Lau, "Recent trends in texture classification: a review", Symposium on Progress in Information & Communication Technology, pp. 63-68, 2009.
- [18] Matteo Masotti and Renato Campanini, "Texture classification using invariant ranklet features", Pattern Recognition Letters, vol. 29, Iss. 14, pp. 1980-1986, 15, October 2008.
- [19] Alireza Tavakoli Targhi, Jan-mark Geusebroek and Andrew Zisserman, "Texture Classification with Minimal Training Images", IEEE International Conference on Pattern Recognition, 2008.
- [20] Di Huang, Caifeng Shan, Mohsen Ardebilian and Liming Chen, "Facial Image Analysis Based on Local Binary Patterns: A Survey", pp. 1-14, 2008.



Ramireddy Obulakonda Reddy received Bachelor's degree in Computer Science and Engineering from Sri Krishnadevaraya University and Master's of Technology degree in Computer Science and Engineering and Pursuing Ph.D, at JNTUA, Anantapur . He is having 10 years of experience in teaching and research and taught for bachelor and master degree courses. His area of interest includes Pattern Recognition, Image Analysis, cloud computing, Network Systems, Software Architecture. He attended many conferences and workshops and communicated papers to reputed National and International journals.



Prof. **B. Eswara Reddy** obtained his Bachelor Degree in Computer Science and Engineering and Master's degree in Computer Science Engineering and Ph.D degree from JNTU University and at present working as an Professor in Department of Computer Science and Engineering, JNTU College of Engineering, JNTU University, Anantapur.He has more than 15 years of teaching and research experience. His area of interest includes Image Processing, Pattern Recognition, Data mining and Cloud Computing. He has published more than 40 publications at national and international level. He is one of the authors of the text book titled "Programming with Java" published by Sanguine/Pearson publications.

He has served as NSS Programme officer, Officer in charge of Academic Section and Officer in Charge of Central Computer Center. He is a fellow of IE (India) and member of ISTE, ISCA, CSI and IEEE.



Prof. **E. Keshava Reddy**, obtained his Master's degree and Ph.D in Mathematics and at present working as an Professor in Department of Mathematics , JNTUA College of Engineering, JNTU University, Anantapur. His areas of interest include Mathematics and Pattern Recognition, Image analysis, Computer networking, data mining, and other latest trends in technology. He has more than 14 years of experience in teaching and research in the area of Mathematics and Computer Science and Engineering. He has attended and conducted many conferences and workshops in different areas and communicated papers to reputed National and International journals. He has served as Controller of Examinations i/c JNTUA Anantapur.