International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2015): 78.96 | Impact Factor (2015): 6.391

The Use of Run Length and Contrast Features with Neural Network for Texture Recognition

Suhair H. S. Al-Kilidar[1], Loay E. George[2]

Baghdad University/College of Science

Abstract: ***The analysis of texture parameters is a useful way of increasing the information obtainable from images. It is an ongoing field of research, whether these images are medical images or natural, with applications to characterizes the variation in textures and classify the texture. In this research, a comparative study of conventional texture-analysis methods and the proposed methods. The textural features extracted by these methods are exploited to classify regions of interest (ROI's). Four different sets of features are proposed: the first set is a simple modified features extracted from the traditional Run Length Matrix (GLCM); the second set is features extracted from Contrast Matrices (CM); the third set uses features extracted from combination between Contrast Matrices (CM) and run length feature vector; the fourth way was passing the extracted sets of these previous proposed set through Artificial Neural Network (ANN) for classification purpose. It proved that the proposed methods are superior to the conventional texture-analysis method with respect to classification accuracy and computational complexity.***

Keywords: Regions of Interest (ROI's), Run Length Matrix (RLM), Contrast Matrices (CM), Artificial Neural Network (ANN)

1. Introduction

Texture analysis is an important and useful area of study in machine vision and image processing. Texture can be described as an attribute representing the spatial arrangement of the gray levels of the pixels in a region of a digital image. [Rus99] had loosely defined image texture as a descriptor of local brightness variation from pixel to pixel in a small neighborhood through an image. Many definitions of texture are presented, some of them are consciously motivated, and others are driven completely by the application in which the definition will be used [Che98]. This makes texture analysis a difficult and challenging problem; it has a rich research area in various fields such as medical image analysis, remote sensing, documents processing, industrial inspection and quality control and image retrieval etc.

Generally, most of new applications involve automatic features extraction from any image which are used in various classification tasks, such as rock classification, fabric classification etc. depending on the particular classification task, the extracted features capture color properties, morphological properties or certain texture properties of the image.

Al-Kadi [Alk08] used three statistical-based (Co-occurrence Matrices , Run Length Matrices and Auto-covariance function) and two model-based methods (Gaussian Markov random field (GMRF) and fractional Brownian motion (fBm)) to extract texture features from eight different texture images to improve the accuracy of texture classification based on extracting texture features. The overall accuracy improved to 96.94% and 96.55% by using GMRF with statistical based methods, such as Gray level co-occurrence (GLCM) and run-length (RLM) matrices; respectively. Nailon [Nai10] used the statistical and fractal texture analysis approaches in the context of medical imaging and provided comprehensive real-world examples, two contrasting methods were presented in the case studies for evaluating the performance of the texture analysis methodologies. In form of two case studies, these approaches used in clinicalpracticeapplications;the first case is the classification of distinct regions in cancer images, which could be developed further towards automatic classification; while in 2ndcasetexture analysis is presented as an objective means of identifying the different patterns of prion protein found in variant CJD (vCJD) and sporadic CJD. Duda et al. [Dud14] proposed to analyze simultaneously triplets of prostate MR images, corresponding to the same prostate slice, but derived from different image series: the contrast-enhanced T1-, T2-, and the diffusion-weighted one. Two classes of prostatic tissue were differentiated: tumorous and healthy. Six different texture analysis methods were used: GLH-, COM-, RLM-, GM-, AC-, and FM-based. Their ability of characterizing prostatic tissue was assessed using three classifiers: Logistic Regression (LR) [Hos13], Neural Network (NN) [Bis95, Rip96] and SVM. The 10-fold cross validation[Dud01] was used to assess the classification accuracies. The best overall classification result exceeded 99% and corresponded to the application of the SVM classifier. Szczypińsk et al. [Szc15] evaluated the effectiveness of identification of barley varieties based on image derived shape, color and texture attributes of individual kernels. Varieties can be determined by means of discriminate analysis, including reduction of feature space dimensionality, linear classifier ensembles and artificial neural networks, with high balanced accuracy ranging from 67% to 86%. This study demonstrated that classification results can be significantly improved by standardizing individual kernel images in terms of their anteroposterior and dorsoventral orientation and performing additional analyses of wrinkled regions.

This paper presents enhanced methods for texture identification using Run Length Matrices RLM and Contrast Matrices CM. These methods could improve system recognition accuracy since they depend on combinations of Contrast Matrices and run length feature, and consequently, the combinations of their corresponding derived features. Also, the classification results in case of using ANN classifier methods instead of Euclidean distance criteria are given.

The next sections are organized as follows: In the next section, an overview of texture analysis methods used in this research is presented by clarifying some of the concepts related to the used methods and the attributes that can be derived from them. Section 3 explains the adopted methodology in this search and the proposed manners for modifying of the traditional methods and the applied combination between the CM and feature vector derived from RLM. In fourth section the attained results are presented and compared with results of some published articles. Finally, in the last section, some of the derived conclusions are presented.

2. Overview

One of the main approaches to texture characterization of image is statistical approach, in this research the most commonly main categories from this approach were used like run length matrices and contrast matrices.

A. Run Length Matrices (RLM)

Run Length Matrix (RLM) is based on probabilities of pixel runs of each possible length, arranged in a certain direction [Gal75]. Four standard directions of pixel runs are considered, θ = 0◦, 45◦, 90◦, or 135◦. A run length matrix, R(θ), has G columns and M rows, where G is the number of image gray levels, and M is the maximum length of pixel run which can exist in an analyzed image region. The element r_{mg} (m = 1,..., M, and g = 0,..., G) of a run length matrix R(θ) is the number of existing pixel runsof a gray level g, having a length of m,and orientedin a direction θ.

From the original run-length matrix $r_{mg}(i, j)$, many numerical texture measures can be computed. The five original features of run length statistics derived by [Gal75] are as follows:

- Short Run Emphasis Inverse Moment (SRE)

$$SRE = \sum_{i=1}^{Ng}\sum_{j=1}^{Nr}\frac{P(i,j)}{j^2} \quad (1)$$

where P(i,j)is the probability of R(I,j)

- Long Run Emphasis Moment (LRE)

$$LRE = \sum_{i=1}^{Ng}\sum_{j=1}^{Nr} j^2 P(i,j) \;/ \sum_{i=1}^{Ng}\sum_{j=1}^{Nr} P(i,j) \quad (2)$$

- Gray-Level Non-uniformity (GLN)

$$GLN = \sum_{i=1}^{Ng}\left(\sum_{j=1}^{Nr} P(i,j)\right)^2 \quad (3)$$

- Run Length Non-uniformity (RLN)

$$RLN = \sum_{j=1}^{Nr}\left(\sum_{i=1}^{Ng} P(i,j)\right)^2 \;/ \sum_{i=1}^{Ng}\sum_{j=1}^{Nr} P(i,j) \quad (4)$$

- Run Percentage (RP): Fraction of Image in Rums

$$RP = \sum_{i=1}^{Ng}\sum_{j=1}^{Nr} P(i,j) \;/ \sum_{i=1}^{Ng}\sum_{j=1}^{Nr} jP(i,j) \quad (5)$$

[Chu90] proposed two new features, as follows, to extract gray level information in the matrix

- Low Gray-Level Run Emphasis (LGRE)

$$LGRE = \sum_{i=1}^{Ng}\sum_{j=1}^{Nr}\frac{P(i,j)}{i^2} \;/ \sum_{i=1}^{Ng}\sum_{j=1}^{Nr} P(i,j) \quad (6)$$

- High Gray-Level Run Emphasis (HGRE)

$$HGRE = \sum_{i=1}^{Ng}\sum_{j=1}^{Nr} i^2 P(i,j) \;/ \sum_{i=1}^{Ng}\sum_{j=1}^{Nr} P(i,j) \quad (7)$$

In a more recent study, [Das91] described another four feature functions following the idea of joint statistical measure of gray level and run length, as follows:

- Short Run Low Gray-Level Emphasis (SRLGE)

$$SRLGE = \sum_{i=1}^{Ng}\sum_{j=1}^{Nr}\frac{P(i,j)}{i^2 . j^2} \;/ \sum_{i=1}^{Ng}\sum_{j=1}^{Nr} P(i,j) \quad (8)$$

- Short Run High Gray-Level Emphasis (SRHGE)

$$SRHGLE = \sum_{i=1}^{Ng}\sum_{j=1}^{Nr}\frac{P(i,j) . i^2}{j^2} \;/ \sum_{i=1}^{Ng}\sum_{j=1}^{Nr} P(i,j) \quad (9)$$

- Long Run Low Gray-Level Emphasis (LRLGE)

$$LRLGE = \sum_{i=1}^{Ng}\sum_{j=1}^{Nr}\frac{P(i,j) . j^2}{i^2} \;/ \sum_{i=1}^{Ng}\sum_{j=1}^{Nr} P(i,j) \quad (10)$$

- Long Run High Gray-Level Emphasis (LRHGE)

$$LRHGE = \sum_{i=1}^{Ng}\sum_{j=1}^{Nr} P(i,j) . i^2 . j^2 / \sum_{i=1}^{Ng}\sum_{j=1}^{Nr} P(i,j) \quad (11)$$

[Das91] tested all eleven features for classifying of a set of cell images and showed that the last four features gave better performance.

These features are all based on intuitive reasoning, in an attempt to capture some apparent properties of run-length distribution. For example, the eight features illustrated in Figure 1 are weighted-sum measures of the run-length concentration in the eight directions, i.e., the positive and negative 0, 45, 90, and 135 directions. The two drawbacks of this approach are: (i) there is no theoretical proof thatthe maximum texture information can be extracted from the run-length matrix, and (ii) many of these features are highly correlated with each other[Tan98].

Also, [Alb00] proposed to use a run length entropy as a texture feature, in this method, the values of the same feature corresponding to different directions of pixel runs can be averaged.

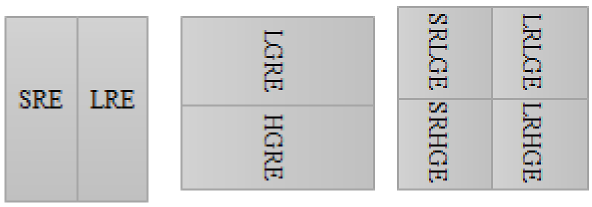

Figure 1: Run-emphasis regions of several traditional run-length texture features

B. Contrast Matrices (CM)

Contrast is the difference in visual properties that makes an object (or its representation in an image) distinguishable from other objects and the background. In visual perception of the real world, contrast is determined by the differences in the color and brightness of the object and other objects within the same field of view. In other words, it is the difference between the darker and the lighter pixel of the image, if it is big the image will have high contrast and in the other case the image will have low contrast [Gon02]. Contrast can be simply explained as a measure for maximum or minimum intensity for the pixel in an image or the difference between the intensity of values surrounding (i.e. the neighborhood of pixel) to each pixel in the image. In this work used 5 manners to compute the Contrast Matrices have been tested:

- The first was to calculate the difference between the pixel value and the total mean of neighborhood values of 3 x 3 pixels (See Figure 2).

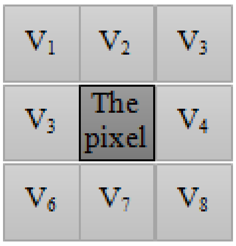

Figure 2: Neighborhood values of 3x3 pixels in gray color.

$$\text{Contrast}_1 = V_{(x,y)} - \overline{V_{(x,y)}} \quad (12)$$

- Second, third, and fourth computed as:

$$\text{Contrast}_2 = Min(A_1, A_2, A_3) \quad (13)$$
$$\text{Contrast}_3 = Max(A_1, A_2, A_3) \quad (14)$$
$$\text{Contrast}_4 = Mean(A_1, A_2, A_3) \quad (15)$$

Where A_1 is the absolute value of the difference between the mean of the values to the left pixel and the mean of the values to the right pixel (See Figure 3a)

$$A_1 = |\overline{V_L} - \overline{V_R}| \quad (16)$$

A_2 is the absolute value of the difference between the mean of the values to the up the pixel and the mean of the values to the down pixel (See Figure 3b)

$$A_2 = |\overline{V_U} - \overline{V_D}| \quad (17)$$

And,A_3 is the absolute value of the difference between the mean of the values to the left diagonal of the pixel and the mean of the values to the right diagonal of the pixel (See Figure 3c).

$$A_3 = |\overline{V_{DL}} - \overline{V_{DR}}| \quad (18)$$

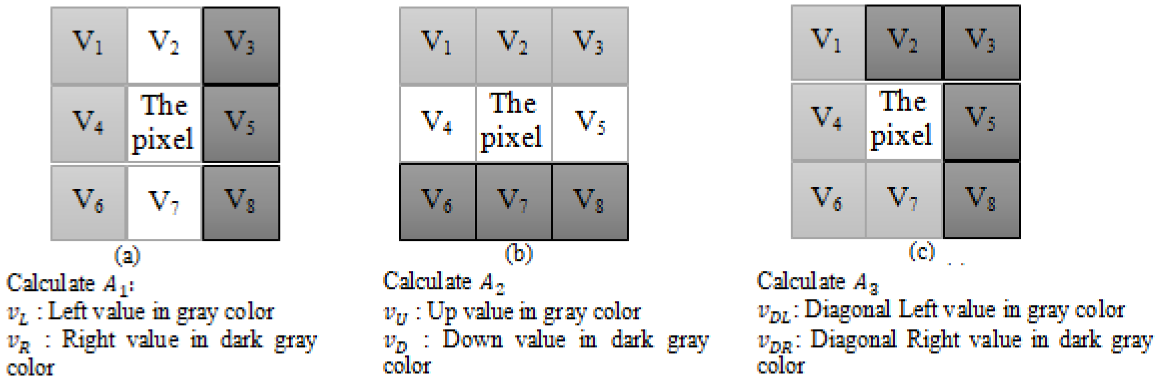

Figure 3: The calculations of A_1 , A_2& A_3

- The fifth manner is calculated as follows: The difference between the sum of the neighborhood values of 3x3 pixels which are larger than the pixel values divided by their number and the sum of the neighborhoods values of 3x3 pixels which are smaller than the pixel value divided by their number

$$\text{Contrast}_5 = Sum_{big} - Sum_{small} \quad (19)$$

Where

$$Sum_{big} = \frac{sum(v) > v}{n_{big}}, Sum_{small} = \frac{sum(v) > v}{n_{small}} \quad (20)$$

C. Recognition Based on Artificial Neural Network (ANN)

In this research ANN was adopted for texture classification, the adopted ANN topology is feed forward network, it consists of three layers (i.e., input layer, output layer, and hidden layer),as depicted in Figure (4). The network architecture is defined by assigning its topological parameters [Abd14B]:

- Number of Input Nodes: The number of input nodes is equal to number of selected features which gave highest classification.

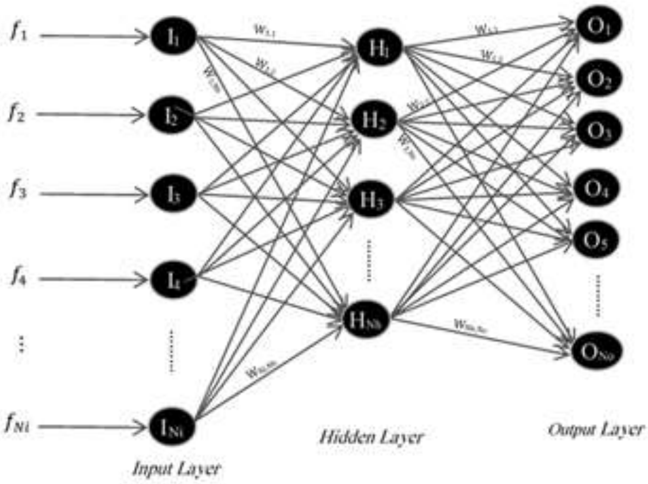

Figure 4: The structure of Feed Forward Neural Network

- Number of Output Nodes: The output layer consists of a number of nodes; equal to the binary number bits needed to the number of texture classes.
- Number of Hidden Nodes: The trial-and error mechanism had been followed to determine the proper number of hidden nodes, because it is difficult to give an accurate formula that can precisely determine the number of hidden nodes, however is presumed a little number of nodes. There are many reasons for this assumption; the first is the computation time; because the network needs

longer time in network training phase whenever the number of hidden nodes is high. The second reason is that using a very large number of hidden nodes leads to high accuracy of training results but a high error ratio in test results.

- Learning Rate: it is a parameter which controls two conflicted requirements: the fast convergence and estimation of stable weights.
- Activation Function: activation functions are mathematical formulae that determine the output of a processing node; each unit takes its net input and applies an activation function to it. Commonly, thenonlinear functions have been used as activation functions which are desirable for network learning.

3. Proposed System

In this study a texture classification system is presented. The system work passes through the following main phases: (i) the enrollment phase and (ii) the classification phase. In the enrollment phase, the system is trained for each class through its texture discriminating features. While in the cognition phase the system works to recognize and classify the texture. The layout of the proposed system is shown in Figure (5); it consists of: (i) color decomposition, (ii) the recognition and classification phase. In the enrollment phase, the system is trained to the each class through its texture discriminating features. While in the cognition phase the system works to recognize and classify the texture. The system consists of: (i) color decomposition, (ii) feature extraction using RLM, CM, and the proposed method, (iii) features analysis and selection. and (iv) matching stage using distance measures or ANN method.

3.1 Preprocessing

Pre-processing stage is the sequence of processing operations which are applied on image (including the loading of image data as file input); these operations make image data appropriate for related information extraction task which can lead to best results; this stage is the first and important stage in any recognition system. In this study the following pre-processes steps were applied:

3.1.1 Color Decomposition

As first step after image data loading, the loaded primary color bands are decomposed into the four color channels (or bands). The basic color components are Red, Green, Blue and the 4^{th} considered band is the Brightness (i.e., Gray); and this gray color value is calculated by applying the equation[Abd14A][Lar16]:

$$Brightness = 0.29 * Red + 0.59 * Green + 0.12 * Blue \quad (21)$$

That is, the color image component are decomposed into four dedicated color bands(i.e., red, green and blue and their corresponding brightness band, such that each band assigned to only one color component.

3.1.2 Quantization

The main reason to apply quantization process is tobypass the main drawback in using big Run Length Matrix (RLM) that is involving with micro-texture details; in such case large memory requirement should fulfilled for storing these matrices and consequently high computational complexity is required. The quantization process removes some of the information details through mapping a group of close data values to single a value.Theused linear equation that mapping each pixel value of band matrix from the range (0 to 255) with a new value (0 to MaxBandVal) is:

$$Quantization = Round\left(MaxBand\,Value * \left(\frac{Gray\ Level}{255}\right)\right) \quad (22)$$

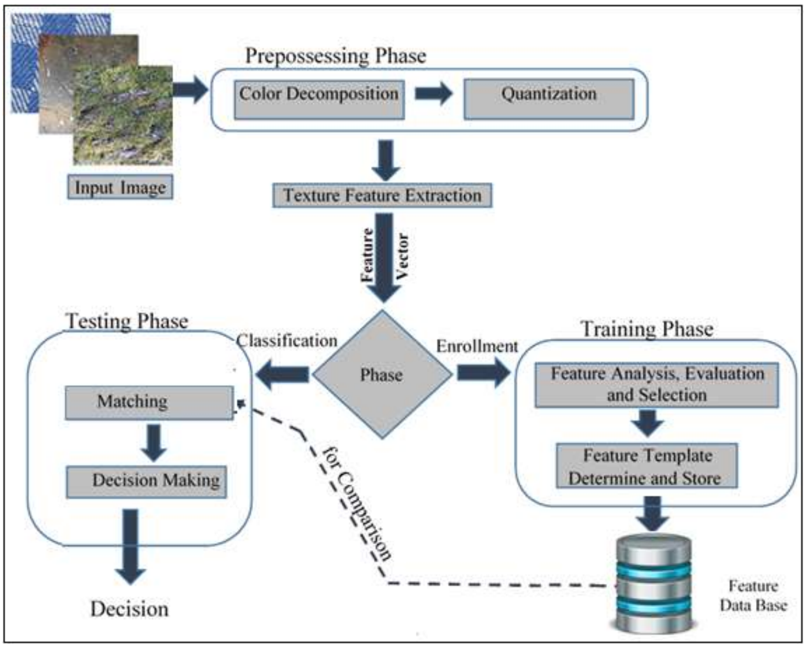

Figure: Block diagram of the proposed Texture classification system

The quantization was applied on the four considered color bands of each image. In this study different MaxBandVal values (8, 16, 20, 24, 28, 32, 40 and 55) were tested to find the best value that leads to best discrimination performance. The test results indicated that the value MaxBandVal =8 led to highest classification results when applied RLM and the value MaxBandVal=40 give the best classification when applied on Gray_ContrastMatrix.

3.2 Feature Extraction

In order to provide accurate classification of textures, the most discriminating information representing the view pattern must be extracted. The best extracted information is gathered in a features vector to distinguish the classes from each others. Many different feature extraction methods have been introduced over the past few decades and they used for texture classification problems as described previously [Alk08, Bhi10, Lin09, Nur13, and Tou09]. In this research, two of the texture analysis methods have been applied to extract feature vector by utilizing the spatial gray Run Length matrices (RLM) and Contrast Matrices (CM). Also, these methods were modified and combined to extract a proper discriminating feature vector, as clarified in the followings:

3.2.1 Gray Level Run Length Method (RLM)

The Run Length Matrices (RLMs) of the used images belong to the downloaded datasets is computed. A set consists of 11 texture features was computed from each determined spatial run length matrix of each color band; the set of features consists ofShort Run Emphasis Inverse Moment (SRE), Long Run Emphasis Moment (LRE), Gray-Level Non-uniformity (GLN), Run Length Non-uniformity (RLN), Run Percentage (RP) (i.e., fraction of image in runsand etc. They are determined using equations (1 to 11); which are explained in sectionIIpart A.

3.2.2 Apply Contrast Method (CM)

We extracted five Contrast Matrices which explained previously, then the histogram is calculated for these matrices, it is the focus of many basic image processing operations. The involved steps of this method lead to redistribute the histogram of the matrix in, order to produce a uniform density [Sap11]. The widening of brightness density is obtained by grouping the adjacent grey values which are close to certain value. Thus, this effect is suitable to discriminate the various regions in image. The histogram is calculated using this code:

```
Code No.1: calculated the histogram of the matrix

 His ← [0 … 510] // array
 For i = 0 to Row Matrix
   For j = 0 to Column Matrix
     v = Round (Matrix [i, j] + 255);
     His[v] = His[v] + 1
   Next j
 Next i
```

Six features from each type of contrast matrix (that previous explained in section II paragraph A), were used after calculating the histogram to the Contrast matrices, these feature explained as below:

$$Moment\ (k) = \sum_{i=0}^{510} His(i)(i-255)^k \quad (23)$$

Where k = 1, 2, 3, 1.5 the moments are called high order moments and when k =0.5, 0.75 the moments are called low order moments.

3.2.3 Application of Modified Methods

The following two types of modifications were applied:

A. The first Variant of the Gray Level Run Length Matrices

In this step, three additional Run Length Matrices (RLMs) were added in addition to the four gray level run length matrices. The traditional RLM are computed along the four directions (0◦, 45◦, 90◦, and 135◦), and the additional three matrices are established as: (i) the mean of the corresponding elements traditional RLM matrices; (2) the Maximum values of the corresponding elements of the four traditional RLM matrices; and(3) the Minimum values of corresponding elements. Then 11 featuresare extracted from these 7 Run Length Matrices (RLM) using the abovementioned features equations from (2.1) to (2.11) in section II part A.The features vector was calculated from 7 run length matrices after color decomposition and quantization by MaxBandValue =8 gray level for each band.

B. Combined Feature Vector Using CMand RLM

The second carried out method is by integrating the Contrast Matrices (CN) with theRLM features calculated from equations (1) to (11). The combined feature vector consists of 55 characteristics for each band.

3.3 Feature Analysis

A training set of samples was used to train the classifier and to address the feature list. While, the test set is used to assess the recognition accuracy of the system (after the training phase). To get a robust recognition performance, there is a need to find out the list of features which shows little intra-class variability. In this work, the selection of these attributes was accomplished due to their inter-class stability. Through the training phase, features were selected from the overall set of features; and the selection was due to the comprehensive tests which were conducted on the training set of samples to find out the best set of features that can be used to yield highest matching score.

3.4 Classification

The matching score is computed according to similarity measure between the features vectors extracted from the input samples with the stored templates representing the classes. The similarity score should be high for samples belong to same class and low for those belong to different classes. The selection of matching samples is usually a difficult pattern recognition task due to large intra-class variations (i.e., due to variations in sample images for the same class) and large inter-class similarity (i.e., due to similarity between samples images from different class).

In this project, the features of samples belong to training set were used to yield the template mean feature vector for each class. The mean feature vector ($\bar{F}$) of each class, and the corresponding standard deviation vector (σ) are determined and saved in a dedicated database during the training phase, it have been used either to match the tested samples data which belong to test set and those used as training samples. Thesemean and standard deviation vectors were used as template vectors. The determination of mean and standard deviation was accomplished using the following equations:

$$\bar{F}(c,f) = \frac{1}{S}\sum_{i=1}^{S} F_i(c,f) \qquad (24)$$

$$\sigma(c,f) = \sqrt{\frac{1}{S}\sum_{i=1}^{S}\left(F_i(c,f) - \bar{F}(c,f)\right)^2} \qquad (25)$$

Where c, f, s are the classes number, feature number and sample number, respectively.

The similarity distance measure for any tested feature (f) is computed using the corresponding template values. The most commonly used similarity measure is Euclidean distance measure (D_1), but, the main weakness of the basic Euclidean distance function is that if one of input features has a relatively large deflection or range of values, then it can overpower the effectiveness the other features. The considered matching problem here is a dynamic behavior; that is every feature may not have similar behaviors like others. So another type of similarity distance Measure (like: D_2, D_3 and D_4) were computed. The results of using these four distance measures were compared and it was noticed that the results of D_4measure are always better than others distance values. So the normalized Euclidean distance (D_4) had been used to decide the similarity between the extracted feature vectors of the tested samples (f_j) vectors and the templates representing the classes [Con80]:

$$D_1(\vec{T_i}, \vec{F_j}) = \sum_{k=1}^{m}\left|t_i(k) - f_j(k)\right| \qquad (26)$$

$$D_2(\vec{T_i}, \vec{F_j}) = \sum_{k=1}^{m}\left(t_i(k) - f_j(k)\right)^2 \qquad (27)$$

$$D_3(\vec{T_i}, \vec{F_j}) = \sum_{k=1}^{m}\left|\frac{t_i(k)-f_j(k)}{\sigma_i(k)}\right| \qquad (28)$$

$$D_4(\vec{T_i}, \vec{F_j}) = \sum_{k=1}^{m}\left(\frac{t_i(k) - f_j(k)}{\sigma_i(k)}\right)^2 \qquad (29)$$

Where $\vec{T_i}$ is the template (mean) of class i, and$\vec{\sigma_i}$is the corresponding standard deviation of class i.

In order to maximize the probability of correct matching classification and minimize the misclassification rate, the efficiency of classification is calculated for each distance using the following equation [Bhi10]:

$$\eta(\%) = \frac{\text{No. of classified samples}}{\text{Total no. of sample}} \times 100\% \qquad (30)$$

3.4.1 Using Combined Features

Incremental comprehensive tests were conducted on the data set for the selecting best combinations belong to the two features sets, using the same previous matching criteria, to get high matching rates. Thus a partial set consists of 30 pairs of combined traits have been chosen from the comprehensive set of attributes. The process of incremental establishment of best discriminating combined feature vectors was conducted with rate of increase was one additional feature at each testing round; it was repeated till no improvement is occurred in the classification rate. It was observed that during the repeated additions some of the features were added many times.

3.4.2 Using Artificial Neural Network (ANN)

The set of training feature vectors was used to train the established ANN; these vectors are extracted from images and saved in feature vector database. The training vectors are used to train a feed forward neural network by adjusting its nodes weights and bias values using back-propagation algorithm. The computed weights and bias values of trained network are, also, registered in the dedicated database. Before feeding the feature vector to the network the value of the involved features must normalized because it is important to unify the dynamic ranges of all involved features. The normalization of feature (f) was performed using the following equation:

$$Nor(f_i) = Round\left(\frac{f_i - \mu_{f_i}}{\sigma_{f_i}}\right)$$

Where $Nor(f_i)$ is the normalized value of feature i, f_i is the i^{th}feature, μ_{f_i} is the mean of i^{th}feature, and σ_{f_i} is the standard deviation of the feature.

In training stage the network starts with a random set of weights and the training sample is presented at the input layer. Then, the outputs of the network are evaluated and compared with the expected "binary" output vector, the error is calculated and the results are fed back from output layer to adjust weights. These training steps are repeated for all training set, and at each time the weights are adjusted. The training continues until one of the stopping conditions is satisfied (i.e., the maximum number of iterations is passed over, or the overall error becomes lower than a predefined target error) and the network is considered ready for decision making tasks.

In the established system, a three layers feed forward neural network architecture had been adopted. The adopted neural network consists of one input layer, one output layer, and one hidden layer. The number of input nodes is set equal to the number of the best combination of discriminating features. the number of hidden nodes was varied to find out the best smallest number of hidden nodes required to lead best recognition; taking into consideration that each additional hidden node causes extra computation during both training and classification phases.

4. Experimental Results

The methods were tested on various color images for three data sets. The samples images have bitmap (BMP) format with color depth 24 bit/pixel; and the size of each image is (128x128) pixels. The sets are Bark, Marble and Fabric, each set consisted of 13 classes and 16 samples into each class. Data set was used in two ways: Firstly, the data of each class was divided into two sets: 11 are used as training set and 5 sample are used as testing set. So the training data size is 143 samples and the testing data size is 65 and the total of samples equal to 208 for each set.Secondly, 8 samples from each class was used as training samples, and 8 samples for testing samples; therefore we have 104 sample for both training group and test group. The used 3 groups of textures sets are taken from Salzburg Texture Image Database (STex); the Salzburg Texture Image Database (STex) is a large collection of 476 color texture image that have been captured around Salzburg, Austria. The images have been selected to be used in texture retrieval experiments and are fairly homogeneous. STex is significantly larger than the VisTex texture image database and more homogeneous than other databases proposed for texture classification research (Outex, A LOT)).

Some of the used samples are presented in Figure (6). At first,a detail comparisons was made between various features of the traditional method (RLM).The attained test results indicated that the best average accuracy of classification is (83.1%). While, the best results obtained from applying the introduced enhanced methods reached around (100%) for the training set and (99.0%) for the testing set (see Table 4, Table 5, and Table 6).

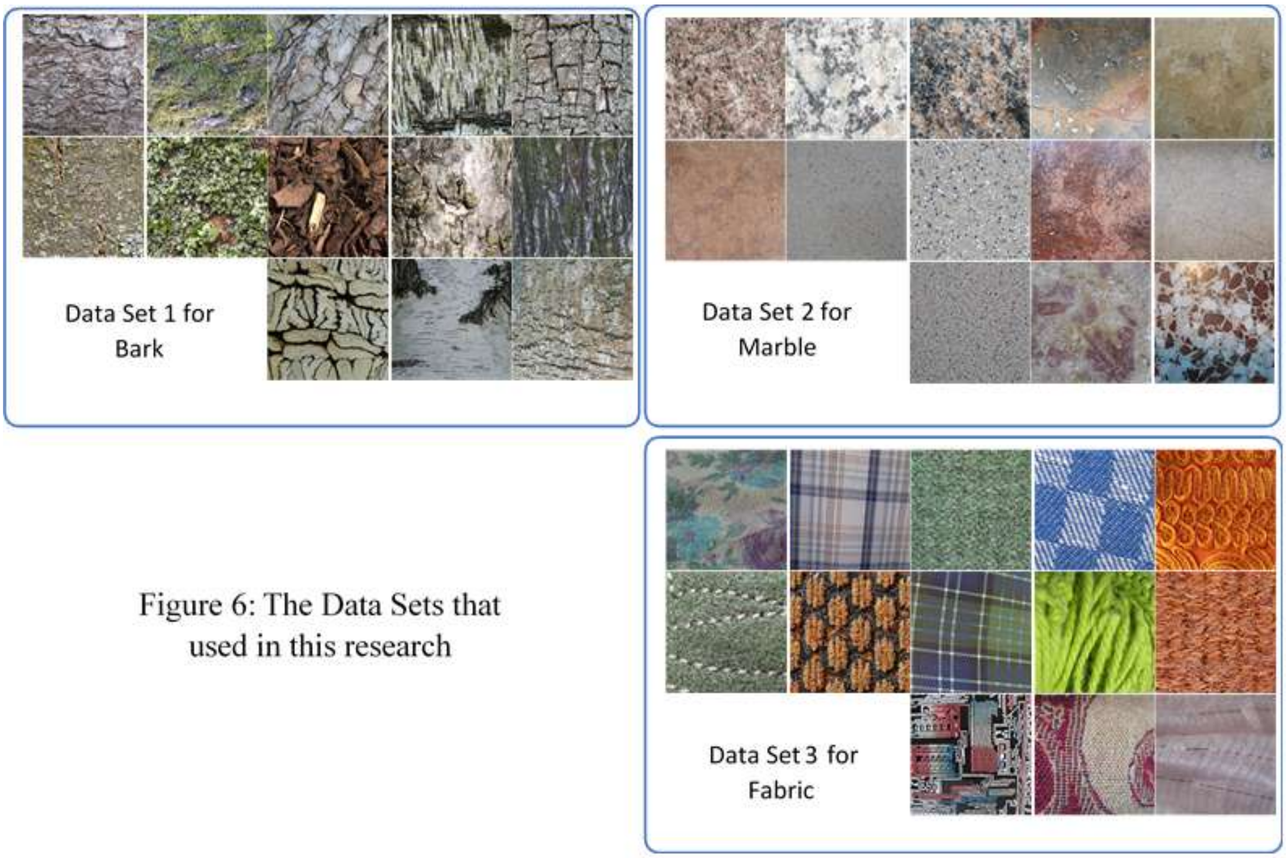

Figure 6: The Data Sets that used in this research

Table 1: Feature Index Table for Run Length Matrices

Feature Name	The Band	Feature Index	Feature Name	The Band	Feature Index
F1: SRIM (Short Run Emphasis Inverse Moment)	Red (90°)	786	F10: SRLGE (Short Run Low Gray-Level Emphasis)	Green (Min)	931
F2: LRIM (Long Run Emphasis Inverse Moment)	Red (0°)	792	F11: HRLGE (High Run Low Gray-Level Emphasis)	Green (45°)	934
F3: GLN (Gray Level Non-uniformity)	Red (0°)	799	F1: SRIM (Short Run Emphasis Inverse Moment)	Blue (90°)	940
F3: GLN (Gray Level Non-uniformity)	Red (Min)	805	F1: SRIM (Short Run Emphasis Inverse Moment)	Blue (Mean)	943
F5: RP (Fraction of Image in Run)	Red (45°)	815	F3: GLN (Gray Level Non-uniformity)	Blue (Mean)	957
F5: RP (Fraction of Image in Run)	Red (135°)	816	F3: GLN (Gray Level Non-uniformity)	Blue (Min)	959
F6: LGRE (Low Gray-Level Run Emphasis)	Red (90°)	821	F4: RLN (Run Length Non-uniformity)	Blue (0°)	960
F7: HGRE (High Gray-Level Run Emphasis)	Red (Max)	832	F4: RLN (Run Length Non-uniformity)	Blue (90°)	961
F8: SRHGE (Short Run High Gray-Level Emphasis)	Red (0°)	834	F4: RLN (Run Length Non-uniformity)	Blue (Max)	965
F8: SRHGE (Short Run High Gray-Level Emphasis)	Red (90°)	835	F5: RP (Fraction of Image in Run)	Blue (90°)	968

F10: SRLGE (Short Run Low Gray-Level Emphasis)	Red (0°)	848	F5: RP (Fraction of Image in Run)	Blue (135°)	969
F11: HRLGE (High Run Low Gray-Level Emphasis)	Red (90°)	856	F7: HGRE (High Gray-Level Run Emphasis)	Blue (0°)	981
F11: HRLGE (High Run Low Gray-Level Emphasis)	Red (45°)	857	F8: SRHGE (Short Run High Gray-Level Emphasis)	Blue (0°)	988
F1: SRIM (Short Run Emphasis Inverse Moment)	Green (45°)	864	F8: SRHGE (Short Run High Gray-Level Emphasis)	Blue (90°)	989
F1: SRIM (Short Run Emphasis Inverse Moment)	Green (Max)	867	F9: LRHGE (Long Run High Gray-Level Emphasis)	Blue (0°)	995
F2: LRIM (Long Run Emphasis Inverse Moment)	Green (135°)	872	F3: GLN (Gray Level Non-uniformity)	Gray (Max)	1035
F3: GLN (Gray Level Non-uniformity)	Green (Min)	882	F3: GLN (Gray Level Non-uniformity)	Gray (Min)	1036
F4: RLN (Run Length Non-uniformity)	Green (90°)	884	F4: RLN (Run Length Non-uniformity)	Gray (0°)	1037
F5: RP (Fraction of Image in Run)	Green (45°)	892	F4: RLN (Run Length Non-uniformity)	Gray (45°)	1039
F8: SRHGE (Short Run High Gray-Level Emphasis)	Green (Mean)	915			

Table 2: Feature Index Table for Contrast Matrices

Feature Name	The Band	Feature Index	Feature Name	The Band	Feature Index
F1: Moment with k=1	Red (contrast 1)	1165	F6: Moment with k=1.5	Green (contrast 1)	1220
F1: Moment with k=1	Red (contrast 4)	1168	F1: Moment with k=1	Blue (contrast 2)	1226
F1: Moment with k=1	Red (contrast 5)	1169	F1: Moment with k=1	Blue (contrast 3)	1227
F2: Moment with k=2	Red (contrast 1)	1170	F2: Moment with k=2	Blue (contrast 1)	1230
F2: Moment with k=2	Red (contrast 5)	1174	F3: Moment with k=3	Blue (contrast 1)	1235
F3: Moment with k=3	Red (contrast 1)	1175	F3: Moment with k=3	Blue (contrast 4)	1238
F3: Moment with k=3	Red (contrast 2)	1176	F3: Moment with k=3	Blue (contrast 5)	1239
F4: Moment with k=0.5	Red (contrast 1)	1180	F4: Moment with k=0.5	Blue (contrast 1)	1240
F4: Moment with k=0.5	Red (contrast 2)	1181	F4: Moment with k=0.5	Blue (contrast 3)	1242
F4: Moment with k=0.5	Red (contrast 3)	1182	F4: Moment with k=0.5	Blue (contrast 4)	1243
F4: Moment with k=0.5	Red (contrast 5)	1184	F5: Moment with k=0.75	Blue (contrast 2)	1246
F5: Moment with k=0.75	Red (contrast 3)	1187	F6: Moment with k=1.5	Blue (contrast 1)	1250
F5: Moment with k=0.75	Red (contrast 4)	1188	F6: Moment with k=1.5	Blue (contrast 2)	1251
F6: Moment with k=1.5	Red (contrast 3)	1192	F6: Moment with k=1.5	Blue (contrast 3)	1252
F6: Moment with k=1.5	Red (contrast 5)	1194	F2: Moment with k=2	Gray (contrast 1)	1260
F1: Moment with k=1	Green (contrast 1)	1195	F3: Moment with k=3	Gray (contrast 5)	1269

Table 3: Feature Index Table for Combined Contrast Matrices and the feature from RLM.

Feature Name	*The Band*	*Feature Index*	*Feature Name*	*The Band*	*Feature Index*
F1: SRE (Short Run Emphasis Inverse Moment)	Red (contrast 3)	1287	F3: GLN (Gray Level Non-uniformity)	Green (contrast 1)	1350
F1: SRE (Short Run Emphasis Inverse Moment)	Red (contrast 4)	1288	F8: SRHGE (Short Run High Gray-Level Emphasis)	Green (contrast 5)	1379
F1: SRE (Short Run Emphasis Inverse Moment)	Red (contrast 5)	1289	F9: LRHGE (Long Run High Gray Level Emphasis)	Green (contrast 5)	1384
F2: LRE (Long Run Emphasis Inverse Moment)	Red (contrast 3)	1292	F1: SRE (Short Run Emphasis Inverse Moment)	Blue (contrast 1)	1395
F3: GLN (Gray Level Non-uniformity)	Red (contrast 2)	1296	F1: SRE (Short Run Emphasis Inverse Moment)	Blue (contrast 3)	1397
F3: GLN (Gray Level Non-uniformity)	Red (contrast 5)	1299	F1: SRE (Short Run Emphasis Inverse Moment)	Blue (contrast 4)	1398
F4: RLN (Run Length Non-uniformity)	Red (contrast 2)	1301	F2: LRE (Long Run Emphasis Inverse Moment)	Blue (contrast 3)	1402
F4: RLN (Run Length Non-uniformity)	Red (contrast 4)	1303	F4: RLN (Run Length Non-uniformity)	Blue (contrast 2)	1411
F5: RP (Fraction of Image in Run)	Red (contrast 2)	1306	F4: RLN (Run Length Non-uniformity)	Blue (contrast 4)	1413
F8: SRHGE (Short Run High Gray-Level Emphasis)	Red (contrast 4)	1323	F5: RP (Fraction of Image in Run)	Blue (contrast 2)	1416
F9: LRHGE (Long Run High Gray Level Emphasis)	Red (contrast 1)	1325	F6: LGRE (Low Gray-Level Run Emphasis)	Blue (contrast 1)	1421
F9: LRHGE (Long Run High Gray Level Emphasis)	Red (contrast 2)	1326	F9: LRHGE (Long Run High Gray Level Emphasis)	Blue (contrast 2)	1436

F9: LRHGE (Long Run High Gray Level Emphasis)	Red (contrast 4)	1328	F11: LRLGE (Long Run Low Gray-Level Emphasis)	Blue (contrast 3)	1447
F11: LRLGE (Long Run Low Gray-Level Emphasis)	Red (contrast 2)	1336	F11: LRLGE (Long Run Low Gray-Level Emphasis)	Blue (contrast 4)	1448
F11: LRLGE (Long Run Low Gray-Level Emphasis)	Red (contrast 4)	1338	F2: LRE (Long Run Emphasis Inverse Moment)	Gray (contrast 1)	1455
F1: SRE (Short Run Emphasis Inverse Moment)	Green (contrast 1)	1340	F3: GLN (Gray Level Non-uniformity)	Gray (contrast 2)	1461
F1: SRE (Short Run Emphasis Inverse Moment)	Green (contrast 5)	1344	F9: LRHGE (Long Run High Gray Level Emphasis)	Gray (contrast 1)	1490
F2: LRE (Long Run Emphasis Inverse Moment)	Green (contrast 1)	1345			

Table 4: The best selected RLM feature which provided highest classification rate; where (A) denotes the case (11 samples for training& 5 samples for testing), (B) denotes thecase (8 samples for training & 8 samples for testing)

Data sets	Quantization	Similarity Distance Measures	Original Feature Length	No. of Selected Features	Index of Feature	Correct classification rate		
						Training Data	Testing Data	Total Data
Set 1 (A)	8	D4	308	13	943, 960, 857, 884, 799, 864, 832, 792, 940, 1039, 856, 867, 934	92.3	93.8	92.7
Set 1 (B)	8	D4	308	6	961, 968, 1018, 821, 1037, 969	96.1	82.6	89.3
Set 2 (A)	8	D4	308	11	882, 969, 988, 816, 931, 1035, 835, 805, 786, 892, 834	97.9	98.4	98.0
Set 2 (B)	8	D4	308	6	882, 965, 915, 995, 815, 872	100	95.1	97.5
Set 3 (A)	8	D4	308	7	968, 989, 805, 792, 786, 1036, 848	99.3	100	99.5
Set 3 (B)	8	D4	308	5	957, 981, 816, 940, 1036	100	91.3	95.6

Table 5: The best selected Contrast Matrices (CM) features which provided highest classification rate; where (A) denotes thecase (11 samples for training& 5 samples for testing), (B) denotes thecase (8 samples for training & 8 samples for testing)

Data sets	Similarity Distance Measures	Original Feature Length	No. of selected Feature	Index of Feature	Correct classification rate		
					Training Data	Testing Data	Total Data
Set 1 (A)	D4	120	13	1181, 1235, 1252, 1239, 1260, 1174 1235, 1168 1235, 1243, 1242, 1268, 1220	91.6	87.6	90.3
Set1(B)	D4	120	12	1246, 1250, 1243, 1165, 1170, 1187, 1195, 1230, 1176, 1174, 1184, 1242	97.1	78.8	87.9
Set 2 (A)	D4	120	8	1181, 1238, 1235, 1174, 1251, 1269, 1260, 1169	95.8	96.9	96.1
Set 2 (B)	D4	120	5	1174, 1182, 1280, 1226, 1168	99.0	94.2	96.6
Set 3 (A)	D4	120	8	1180, 1243, 1194, 1170, 1227, 1240, 1165, 1187	97.9	100	98.5
Set 3 (B)	D4	120	5	1192, 1239, 1188, 1175, 1254	100	91.3	95.6

Table 6: The best selected Contrast Matrices (CM) and the feature extracted from Run Length Matriceswhich provided highest classification rate; where (A) denoted thecase (11 samples for training& 5 sample for testing), (B) denotes thecase (8 sample for training & 8 sample for testing)

Data sets	Quantization	Similarity Distance Measures	Original Feature Length	No. of Selected Feature	Index of Feature	Correct classification rate		
						Training Data	Testing Data	Total Data
Set 1 (A)	40	D4	220	10	1338, 1490, 1398, 1287, 1350, 1328, 1436, 1338, 1384, 1436	93.4	96.9	94.4
Set 1 (B)	40	D4	220	9	1299, 1395, 1379, 1326, 1421, 1402, 1455, 1340, 1344	99.0	83.6	91.3
Set 2 (A)	40	D4	220	6	1296, 1397, 1338, 1447, 1306, 1325	96.5	100	97.5
Set 2 (B)	40	D4	220	5	1289, 1411, 1338, 1413, 1301	100	93.2	96.6
Set 3 (A)	40	D4	220	8	1292, 1448, 1461, 1323, 1345, 1416, 1288, 1398	98.6	98.4	98.5
Set 3 (B)	40	D4	220	4	1303, 1448, 1336, 1328	100	95.1	97.5

Deciding the proper number of nodes in the hidden layer is a very important part of deciding the overall neural network architecture. Trial-and-error mechanism was adopted in this research work to find out the proper number of hidden nodes. Various values of hidden nodes numbers were tested to examine their effect on ANN performance. Tables (7),(8) and (9) shows the effect of the number of hidden nodes in the hidden layer of the ANN and their effect on classification accuracy rate.

Table 7: The Effect of "Number of Hidden Nodes" on Classification Accuracy Rate for the three texture sets

Data Sets	RLM			CM			Proposed Method		
	No. of Hidden Nodes	Min Training Error	Classification Accuracy Rate	No. of Hidden Nodes	Min Training Error	Classification Accuracy Rate	No. of Hidden Nodes	Min Training Error	Classification Accuracy Rate
Set 1 (Bark)	17	0.10043	96.1	8	0.8512999	99.0	7	0.07713	97.1
	21	0.03126	99.0	14	0.0836545	96.6	15	0.05258	97.5
	22	0.07433	98.5	16	0.0734709	99.5	17	0.03820	98.0
	30	0.03780	99.0	19	0.0603406	97.1	20	0.05383	98.0
Set 2 (Marble)	17	0.03597	99.0	8	0.3440441	77.4	9	0.07844	97.1
	19	0.02363	99.5	13	0.2950156	76.9	10	0.06285	98.0
	22	0.03944	100	19	0.3511264	77.4	13	0.04285	98.5
	29	0.01574	100	20	0.2844794	77.4	19	0.03233	99.0
Set 3 (Fabric)	13	0.04733	99.5	8	0.2646555	80.7	8	0.08921	98.5
	14	0.01890	100	12	0.2403475	88.9	9	0.04412	99.0
	19	0.01716	100	17	0.1468720	89.9	18	0.01672	99.5
	21	0.01650	100	20	0.1240415	94.7	20	0.01766	100

5. Conclusions

In this paper, a set of modified RLM methods are introduced; these methods are improved variants to the traditional Run Length Matrices (RLM). The best recognition rates of the proposed method was (%99) for classification accuracy rate when using the modified Run length Matrices features';while when using the Contrast Matrices (CM); the best attained recognition rate had reached (%98) for classification accuracy rate, and when using the combined feature vector using contrast matrices (CM) and thefeaturesof Run Length Matrices the classification accuracy rate reached (%99.5). The attained rates are better than those reached by the traditional method. For the case of using ANN the attained recognitionrate was (%100) for recognition accuracy rate.

References

[1] [Abd14A] Abdul-Wadood, D. N., George, L. E., & Rasheed, N. A. "Diagnosis of Skin Cancer Using Image Texture Analysis." International Journal of Scientific & Engineering Research Vol. 5, no. Issue 6: Pp. 155-162. 2014.

[2] [Abd14B] Abdul-Wadood, D. N., George, L. E., & Rasheed, N. A. "Texture Classification Using Artificial Neural Network for Diagnosis of Skin Cancer." Integrated Journal of British Vol. 1, no. issue 2: Pp. 20-25. 2014.

[3] [Alb00] Albregtsen, F., Nielsen, B., Danielsen, H.E.; "Adaptive Gray Level Run Length Features From Class Distance Matrices". Vol. 3, in Proc. 15th Int. Conf. on Pattern Recognition, 738-741. 2000.

[4] [Alk08] Al-kadi, O. S.; "Combined Statistical and Model Based Texture Features for Improved Image Classification". In: 4th International Conference on Advances in Medical, Signal & Information Processing. Santa Margherita Ligrue ,Italy, 2008.

[5] [Bhi10] Bhiwani, R. J.; Khan, M. A.; Agrawal, S. M.;. ""Texture Based Pattern Classification". International Journal of Computer Applications Vol. 1, no. No. 1: Pp. 54-56, 2010.

[6] [Bis95] Bishop, C. M. "Neural Networks For Pattern Recognition". New York, USA: Oxford University Press, 1995.

[7] [Che98] Chen, C. H.; Pau, L. F.; and Wang, P. S. P.; "The Handbook of Pattern Recognition and Computer Vision". Second Edition. World Scientific Publishing Co., 1998.

[8] [Chu90] Chu, A. , Sehgal,C. M. and Greenleaf, J. F. Use Of Gray Value Distribution of Run Lengths for Texture Analysis. Vol. 12, in Pattern Recognit, by Lett., 415–420. June 1990.

[9] [Con80] Conners, R.W., Harlow, C.A.;"Toward A Structural Textural Analyzer Based On Statistical Methods". Vol. 12, 224-256. Comput. Vision Graph, 1980.

[10] [Dud14] Duda, D. "Texture Analysis As A Tool For Medical Decision Support. Part 1: Recent Applications For Cancer Early Detection". (Advances in Computer Science Research ,Atlantes Press) vol. 11: Pp. 61-84, 2014.

[11] [Dud01] Duda, R. O., Hart, P. E., & Stork, D. G. "Pattern Classification" . Second Edition. Vol. 58. New York: John Willey and Sons, INC, 2001.

[12] [Gal75] Galloway, M. M. "Texture analysis using gray level run lengths". Computer Graphics and Image Processing. vol.4: page 172–179, June 1975.

[13] [Gon02] Gonzalez,R.C.; Woods, R.E. "Digital Image Processing. Second edition". Prentic Hall, 2002.

[14] [Hos13] Hosmer Jr, D. W., Lemeshow, S., & Sturdivant, R. X. " Applied Logistic Regression". Third Edition. Hoboken, New Jersey, US: John Wiley & Sons, Inc., 2013.

[15] [Nai10] Nailon, W. H. "Texture Analysis Methods for Medical Image Characterisation". In Biomedical imaging. InTech, 2010.

[16] [Rus99] Russ, J. C. "The image processing handbook." CRC press, 1999.

[17] [Sap11] Sapana S. Bagade, Vijaya K. Shandilya. "Use of Histogram Equalization in Image Processing for Image Enhancement".International Journal of Software Engineering Research & Practices Vol. 1, no. 2: Pp. 6 – 10, 2011.

[18] [Szc15] Szczypiński, P. M., Klepaczko, A., & Zapotoczny, P. "Identifying Barley Varieties by Computer Vision".Computers and Electronics in Agriculture 110: Pp. 1-8, 2015.

[19] [Tan98] Tang, X. "Texture Information in Run-Length Matrices." IEEE Transactions on Image Processing Vol. 7, no. Issue. 11: Pp. 1602-1609. 1998.

[20] [Tou09] Tou, J. Y.; Tay, Y. H.; and Lau, P. Y.;. "Recent Trends in Texture Classification: A Review".Symposium on Progress in Information & Communication Technology. Kuala Lumpur, Malaysia, 2009.