

Textural Features and Feature Selection in KAN-NADA Character Recognition

B N Ajay, C Naveena

Abstract- This paper describes an OCR system for printed text documents in Kannada, a South Indian language. Many commercial OCR systems are now available in the market, but most of these systems work for Roman, Chinese, Japanese and Arabic characters. There are no sufficient number of works on Indian language character recognition especially Kannada. In this work we proposed kannada character recognition system using texture features. Here we fuse the texture features like Local Binary Pattern, Local Binary Pattern Variance, Gray Level Local Texture Pattern, Gabor Filter Response and Wavelet Decomposition using concatenation rule and select discriminative texture features by employing wrapper feature selection methods. Finally, K-NN classifier is explored for the purpose of classification. In addition, we also explore the K-NN classifier with different distance functions. This method is simple to implement and realize, also it is computationally efficient.

Keywords: LTP, GLTP, KNN, SFS, SFFS

I. INTRODUCTION

Optical character recognition (OCR) lies at the core of discipline of pattern recognition where the objective is to interpret scanned images of handwritten or typewritten text into computer processable format. Due to the advancement of digital computers many tools are available for word processing and large amount of paper documents are processed for various reasons all over the world. Extracting information from paper documents manually would add significant cost in terms of human labour. So, making computers able to read would allow for substantial savings in terms of the costs for data entry, mail processing, form processing and many other similar situations. This real world problem can be made easy by OCR. Character recognition can be divided into two major categories according to mode of the document given to it i) Online character recognition ii) Offline character recognition. Further, Offline character recognition is divided into i) Printed character recognition ii) Handwritten character recognition. Kannada, the official language of the south Indian state of Karnataka, is spoken by about 48 million people. The Kannada alphabets were developed from the Kadamba and Chalaukya scripts, descendants of Brahmi which were used between the 5 th and 7 th centuries A.D. The basic structure of Kannada script is distinctly different from the Roman script. Unlike many north Indian languages, Kannada characters do not have shirorekha (a line that connects all the characters of any word)

Revised Manuscript Received on December 22, 2018.

First Author name, His Department Name, University/ College/ Organization Name, City Name, Country Name.

Second Author name, His Department Name, University/ College/ Organization Name, City Name, Country Name.

Third Author name, His Department Name, University/ College/ Organization Name, City Name, Country Name.

and hence all the characters in a word are isolated. This creates a difficulty in word segmentation. Kannada script is more complicated than English due to the presence of compound characters. However, the concept of upper/lower case characters is absent in this script. Hence in this work we investigate the suitability of texture features in designing a system for kannada character classification. Character is segmented using a threshold based method and texture features viz., Local Binary Pattern, Local Binary Pattern Variance, Gray Level Local Texture Pattern, Gabor Filter Response and Wavelet Decomposition. These features are used for training and classification using K-Nearest Neighbor Classifier.

II. RELATED WORK

For the past few decades, intensive research has been done to solve the problem of printed and handwritten character recognition. Various approaches have been proposed to deal with this problem. Challenging problems are being encountered and solutions to these are targeted in various ways to improve accuracy and efficiency. A brief overview of the numeral recognition work done on Indian scripts is presented below. M. Hanmandlu et al.,[1] proposed a method for recognition of handwritten Hindi numerals, written in Devanagari script. The method is based on a kind of exponential membership function fitted to the fuzzy sets derived from features consisting of normalized distances obtained using the box approach. The exponential membership function is modified by two structural parameters that are estimated by optimizing the entropy subject to the attainment of membership function to unity. The overall recognition accuracy is found to be 96%. However, the experiments are carried out on a limited database with size of 3500 samples. Rajashekaradhya et al.,[2] used zone centroid and image centroid based distance metric feature extraction method for recognition of Kannada, Telugu, Tamil and Malayalam numerals. The character centroid is computed and the numeral image is further divided into n equal zones. Average distance from character centroid to the each pixel present in zone is computed. Similarly zone centroid is computed and average distance from the zone centroid to each pixel present in zone is computed. Nearest neighbour and feed forward back propagation neural network classifiers are used for subsequent classification and recognition purpose. In [3], U Pal et al. have proposed a modified quadratic classifier based scheme towards the recognition of off-line handwritten numerals of six popular Indian scripts.



One of the scripts is Kannada. The features used in the classifier are obtained from the directional information of contour points of the numerals. For feature computation, the bounding box of a numeral is segmented into blocks and the directional features are computed in each of the block. These blocks are then down sampled by a Gaussian filter and features obtained from the down sampled blocks are fed to modified quadratic classifier for recognition. A five-fold cross validation technique has been used for result computation and they have reported 98.71% accuracy for Kannada scripts obtained by performing experiments on their own data set. Benne et al. [4] proposed recognition system for isolated handwritten numerals recognition for three popular Indian scripts namely, Kannada, Devanagari and Telugu numeral sets. The proposed method is thinning free and without size normalization. The structural features viz. directional density of pixels, water reservoirs, maximum profile distances, and fill hole density are used for handwritten numerals recognition. A Euclidian distance criterion and K-nearest neighbour classifier is used to classify the handwritten numerals. A limited data set consisting of 5250 numeral images are considered for experimentation, and the overall accuracy of 95.40%, 90.20%, and 98.40% is reported for Kannada, Devanagari and Telugu numeral respectively. Dinesh Acharya and et al. [5] have used 10-segment string concept, water reservoir, horizontal and vertical strokes, and end points as features and k-means to classify to classify the Kannada handwritten numeral. They have reported the overall recognition accuracy of 90.5%. G.G. Rajput et al. [6] described a system for isolated Kannada handwritten recognition using image fusion method. Several digital images corresponding to each handwritten numeral are fused to generate patterns, which are stored 8x8 matrices, irrespective of the size of images. The numerals to be recognized are matched using nearest neighbour classifier with each pattern, and the best match pattern is considered as the recognized numeral. The average recognition rate of 95.62% is reported by them obtained by performing experiments on the data set generated locally. In [7], handwritten Kannada character recognition system based on spatial features is proposed. Directional spatial features viz stroke density, stroke length and the number of strokes are employed as features to characterize the handwritten Kannada vowels using K-NN classifier. The average recognition accuracy of 90.1% is reported for vowel characters. In [8], moment based features are used for recognition of Kagunita (the Kannada compound characters resulting from the consonant and vowel combination). These features are extracted using Gabor wavelets from the dynamically pre-processed original image. Multi-Layer Perceptron with Back Propagation Neural Networks are employed for character classification. Average recognition rate of 86% is reported for vowels and for consonants the average recognition reported is 65%. From the work reported in the literature, we note the following. In many of the proposed methods, the dataset size is small in terms of number of samples used for each character/numeral. There is no clear explanation about training and test images in case of supervised algorithms used [9]. Experiments are not being performed on entire

character set of the script/language, eg. not all 49 characters of Kannada script are included.

III. PROPOSED METHOD

The proposed method has training and classification phases. In training phase, from a given set of training images the texture features (LTP/ Gabor/GLTP/WD) are extracted and later feature selection methods like sequential forward selection (SFS), sequential floating forward selection (SFFS), sequential backward selection (SBS) and sequential floating backward selection (SFBS) is used reduce the features. Final the system is trained using the K- nearest neighbor classifier. In classification phase a given test is segmented and then the above mentioned texture features are extracted for classification. These features are queried to K-nearest neighbor classifier to label an unknown Character.

3.1.1 FILTERING

Character data might have been corrupted due to jitters, redundant and extraneous wild points due to erratic pen movements. In addition to this, the hardware limitations in the data collection equipment may also introduce noise to the raw data. So, it is essential to isolate the collected data from these redundant noise components to improve the performance of the recognition system. In practice, a moving average filter or a Gaussian filter is commonly used to remove these noise segments. But, a careful choice of a filter is necessary in order to avoid the loss of important structural features like cusps, dots, etc. In the present work, a Gaussian filter with a sliding window size of seven [10] is used to filter the noisy images.

3.1.2 Morphologica Processing

Binary images contain numerous imperfections. In particular, the binary regions produced by simple thresholding are distorted by noise. Morphological image processing pursues the goals of removing these imperfections by using morphological erode and dilate operations as shown in the below algorithm [11]. Algorithm

- 1) Define a square structuring element of size 3
- 2) Erode the image m number of times to removespur.
- 3) Dilate the image m+n number of times to fill the gaps in the edges and restore the structure.
- 4) Erode the image n number of times to restore the image

3.2 Texture Based Proposed Model

The proposed texture based model consists of four stages: feature extraction, feature level fusion, feature selection and classification. Prior to the feature extraction stage, we required applying the segmentation process.

3.2.1 Feature Extraction

The roughness is reflected by transitions in intensity levels on the surface of a Character in the form of uniform and non-uniform patterns. To exploit this, we recommend extracting texture features from gray scale images of segmented characters using the various texture based models viz., LBP, LBPV, GLTP, Gabor and Wavelets which are explained in the following subsections.

3.2.1.1 Local Binary



Pattern(LBP)

The LBP is a gray-scale and rotational invariant texture operator which characterizes the spatial structure of the local image texture [12]. The gray-scale invariance is achieved by assigning a unique pattern label to every pixel in an image depending on binary pattern generated by comparing its value with those of its neighborhoods. A pattern label is computed by

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c) 2^p$$

$$\text{where, } s(g_p - g_c) = \begin{cases} 1, & (g_p - g_c) \geq 0 \\ 0, & (g_p - g_c) < 0 \end{cases}$$

Here, g_c is the gray value of the central pixel of circularly symmetric neighborhood g_p ($p = 0, 1, \dots, P-1$), g_p is the gray value of its neighbors, P is the number of neighbors and R is the radius of the neighborhood.



Figure 2: Illustration of preprocessing stages of character Ja. (a) Original Image (b) Binarization (c) Filtered and Morphological Image sampled data sample. (d) Size normalized data sample.

3.2.1.2 Wavelet Decomposition (WD)

We performed a single level wavelet decomposition on matrices of each channels of original RGB input image of character using Haar wavelet. It results into four coefficients matrix for each channel. They are approximation matrix coefficients (cA) and three detail coefficient matrices called horizontal (cH), vertical (cV) and diagonal (cD). To analyse the function of wavelet, we compute the reconstructed coefficients matrix for each of the details coefficients matrices of each channel. Then we extract features by calculating energy for each obtained reconstructed coefficients matrix. Since we obtained three reconstructed coefficients matrix for each channel, the number of features is 9.

3.2.1.3 Gabor Filter Response(GFR)

Frequency and orientation representations of the Gabor filter are similar to those of the human visual system. Therefore, we exploited Gabor filter response to represent and discriminate texture of characters. In our proposed model, RGB image is

$$g(x, y, \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x'}{\lambda} + \psi\right)$$

transformed into a gray image for further processing. Gabor filter is a linear filter whose impulse response is 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 and it is given by

where $x' = x \cos \theta + Y \sin \theta$ and $y' = x \sin \theta + y \cos \theta$ represents the wavelength of the cosine factor, θ represents the orientation of the normal to the parallel stripes of a Gabor function, ψ is the phase offset, σ (sigma) is the Gaussian envelope and γ is the spatial aspect ratio specifying the ellipticity of the support of the Gabor function. A filter bank of Gabor filters with various scales and rotations is created. In this work we have considered scales of 0, 2, 4, 8 and 10 and, orientations of 0, 45, 90 and 135. For each obtained response image we extract first three moments as features and thus for an image we have totally 72 features.

3.2.1.4 Local Binary Pattern Variance(LBPV)

In LBPV, the variance of the local region is embedded into the LBP histogram resulting in a LBPV histogram [13]. The LBPV histogram is computed as



3.2.1.5 Gray Level Local Texture Pattern(GLTP)

The LBP model is computationally efficient but inadequate to represent a local region whereas the texture spectrum (TS) model will reveal more textural information but it is computationally burdensome. The GLTP was developed by combining the advantages of the TS and the LBP (Ojala et al., [12]). The GLTP is computationally acceptable and is robust against variations in the appearance of the texture to meet the real world applications [14]. These variations may be caused by uneven illumination, different viewing angles and resolving power of the sensor system. Since it is a rotational invariant and gray-scale shift invariant, it is robust against variable illumination. The GLTP model detects the number of transitions or discontinuities in the circular presentation of texture patterns in a small region, thus it suits to our dataset. When such transitions are found to follow a rhythmic pattern, they are recorded as uniform patterns and are assigned with unique labels. All other non-uniform patterns are grouped under single category. It assigns a GLTP label to every pixel in an image depending on uniformity of pattern around the pixel. This labeled image is represented using a one dimensional histogram with abscissa indicating the GLTP label and ordinate representing its frequency. The following rotational and gray-scale invariant GLTP operator [14] is used for describing a local image texture. here, Δg is a small positive value that represents a desirable gray value and has its importance in forming the uniform patterns. Uniform measure (U) corresponds to the number of spatial transitions in a circular direction to form a pattern string and is de-

$$GLTP_{p,R}^{unif} = \begin{cases} \sum_{p=1}^P s(g_p, g_c) & \text{if } U \leq 3 \\ P \times 9 + 1 & \text{otherwise} \end{cases}$$

$$\text{where, } s(g_p, g_c) = \begin{cases} 0 & \text{if } g_p < (g_c - \Delta g) \\ 1 & \text{if } (g_c - \Delta g) \leq g_p \leq (g_c + \Delta g) \\ 9 & \text{if } g_p > (g_c + \Delta g) \end{cases}$$

$$p = 1, 2, \dots, P$$

defined as

$$U = f(s(g_p, g_c), s(g_1, g_c)) + \sum_{p=2}^P f(s(g_1, g_c), s(g_{p-1}, g_c))$$

$$\text{where, } f(X, Y) = \begin{cases} 1 & \text{if } |X - Y| > 0 \\ 0 & \text{otherwise} \end{cases}$$

For example, GLTP with R = 1 and P = 8 could generate 46 unique labels. For U=0 there exist 3 GLTP labels (0, 8 and 72), for U=2 there are 21 GLTP labels (1 to 7, 9, 16, 18, 24, 27, 32, 36, 40, 45, 48, 54, 56, 63 and 64) and for U = 3 there exist another 21 GLTP labels (10 to 15, 19 to 23, 28 to 31, 37 to 39, 46, 47 and 55). All other non uniform patterns are grouped

$$H(k) = \sum_{k \in [0, K]} \sum_{i=1}^N \sum_{j=1}^M w(LBP_{p,R}(i, j), k)$$

$$\text{where, } w(LBP_{p,R}(i, j), k) = \begin{cases} VAR_{p,R}(i, j) & \text{if } LBP_{p,R}(i, j) = k \\ 0 & \text{otherwise} \end{cases}$$

$$\text{where, } VAR_{p,R} = \frac{1}{P} \sum_{p=0}^{P-1} (g_p - u)^2$$

$$\text{here, } u = \frac{1}{P} \sum_{p=0}^{P-1} g_p$$

under one label 73. Since there are few holes in the GLTP labeling scheme, they are relabeled to form continuous numbering from 1 to 46 using a lookup table. The GLTP labels for some pattern strings are shown in Table 1.

IV. FEATURE LEVEL FUSION

Feature level fusion refers to combining different feature vectors that are obtained by employing multiple feature extraction algorithms. When the feature vectors are homogeneous, a single resultant feature vector can be obtained as a weighted average of the individual feature vectors. When the feature vectors are non homogeneous, we can concatenate them to form a single feature vector [15]. The extracted feature vectors of LBP, LBPV, GLTP, Gabor response and Wavelets are fused in all possible combinations by concatenating the feature vectors. The fused feature vectors are normalized using min-max method.

1) 4.1 Feature Selection

Feature selection is the process of selecting a subset of relevant features for building robust learning models. Feature selection is broadly classified into two categories such as filter model and wrapper model. The filter model relies on general characteristics of the training data to select some features without involving any learning algorithm. The wrapper model requires one predetermined learning algorithm in feature selection and uses its performance to evaluate and determine best features for selection. A well known filter method Relief [16] relies on relevance evaluation. Time Complexity of Relief for a dataset with M instances and N features is O(MN). However, the Relief method does not help to eliminate redundant features. Empirical evidence from feature selection literature shows that, along with irrelevant features, redundant features also affect the speed and accuracy of learning algorithms and thus should be eliminated as well [17]. Therefore, we have exploited feature selection methods based on wrapper model such as sequential forward selection (SFS), sequential floating forward selection (SFFS), sequential backward selection (SBS) and sequential floating backward selection

(SFBS) [18]. The criterion employed in these methods is the correct classification rate of the Bayes classifier assuming that the features obey the multivariate Gaussian distribution. These methods eliminate irrelevant features as well as redundant features but they are computationally



slightly expensive than the filter method.

V. CLASSIFICATION

In this model, the K-NN classifier based on G-statistic, Chi-square and Euclidean distance measure has been used to classify kannada character images.

1.1 DATASETS

In this work we have created our own database despite of existence of other databases as these are less intra class variations or no change in view point. We collected Kannada character images from World Wide Web in addition to taking up some photographs of Characters that can be found in and around. The images are taken to study the effect of the proposed method with large intra class variations. Fig. 3 shows a sample image of each different class. It is clearly understandable that there is a large intra class variation. The large intra-class variability and the small inter-class variability make this dataset very challenging.

1.2 EXPERIMENTATION

In the first part of experimentation, we conducted experiments for the proposed model based on individual texture models. During experimentation, we conducted four different sets of experiments. In the first set of experiment, we used 30% of the samples of each class of a dataset to create class representative vectors (training) and the remaining 60% of the samples for testing purpose. In the second set, third set and fourth set of experiments, the number of training and testing samples is in the ratio 40:60, 50:50 and 60:40 respectively. In each set of experiment, experiments are repeated 20 times by choosing the training samples randomly. As measure of goodness of the proposed model based on individual texture model, we computed minimum, maximum, average and standard deviation of classification accuracy of all the 20 trails using the K-NN classifier. Classification accuracy of the proposed model based on individual texture models (LBP, LBPV, GLTP, GFS and WD) for 30% training, 40% training, 50% training and 60% training are tabulated in figure 4, figure 5, figure 6 and figure 7 respectively. It is observed from figures that the GLTP texture model has achieved a better average classification accuracy in experiment 4 (60% training samples) when compared to other texture models. It is also observed that the proposed model has achieved a good classification accuracy for all individual texture models in experiment 4 (60% training samples) when compared to experiment 1 (30% training samples), experiment 2 (40% training samples) and experiment 3 (50% training samples). Therefore, we present experimental results obtained for 60% training only for fusion of texture features and also for fusion of texture features and application of feature selection method. In second part of experimentation, we conducted experiments based on fusion of texture features and the application of feature selection method. That is, we applied the wrapper feature selection method (SFS, SBS, SFFS and SFBS) on fused texture feature vector to reduce the dimension of feature matrix and to obtain discriminative texture features. During experimentation, experiments are repeated 5 times by choosing the training samples randomly. Classification accuracy of the proposed model based on fusion of texture features and feature selection method for 2

features at a time, 3 features at a time, 4 features at a time and all 5 features are tabulated respectively in Table 2, Table 3, and Table 4. Here we presented results of G-statistic for 60% training as it was observed to have good results for the G-statistic based K-NN classifier. It is observed from Table 2, Table 3 and Table 5 that the fusion of GLTP and WD features with SBS feature selection method has achieved best average classification accuracy when compared to other combinations.

5.3 Discussion

From the experimental results, it is observed that the GLTP texture model has dominant features when compared to LBP, LBPV, Gabor response and Wavelet decomposition. Since the GLTP is build by the advantages of TS and LBP, it reveals more local texture information when compared to texture models such as Gabor response and Wavelet decomposition. The GLTP assigns a label (uniform label or non-uniform label) for each pixel in an image based on the uniformity or non-uniformity of neighborhood, whereas the GFR is based on frequency and orientation of edge information. Though, the GFR is rotation invariant local texture information, fusion of Gabor response with the other texture models such as LBP, LBPV, GLTP and Wavelet decomposition has deteriorated the performance because the Gabor response will not represent edge information in the form of uniform patterns and non-uniform patterns. Fusion of LBPV with other texture models such as LBP, GLTP, Gabor response and Wavelet decomposition has also deteriorated the performance because global information such as variance is embedded with local information in the LBPV. Therefore, the LBPV will deteriorate the classification accuracy. Feature selection after fusion has improved the classification results for all combination of fusion of texture models. In all feature selection methods, the dominant features are selected. When analyzed we observed that, the SBS method on fusion of GLTP and Wavelets has selected only 15 features out of 55 features (GLTP – 46 features and Wavelets – 9 features). Out of 15 features, 10 features are from the GLTP and 5 features are from the WD. Therefore, the GLTP has more number of discriminating features with WD features and improve the classification accuracy. Similarly, the SFS on fusion of GLTP and LBP has selected only 6 features out of 56 features (GLTP – 46 features and LBP – 10 features). All 6 features are from the GLTP alone. This indicates that the LBP has no discriminating dominant features when it is with the GLTP features. Further, the SFBS on fusion of GLTP, LBP and WD has selected only 8 features. Out of 8 features, 4 features are from the GLTP and 4 features are from the WD. No discriminating features of LBP are selected when they are with GLTP and WD features.

The above observations make it clear that the GLTP and WD features have more discriminating and dominating features when compared to the other texture models such as LBP, LBPV and Gabor response.

VI. CONCLUSION

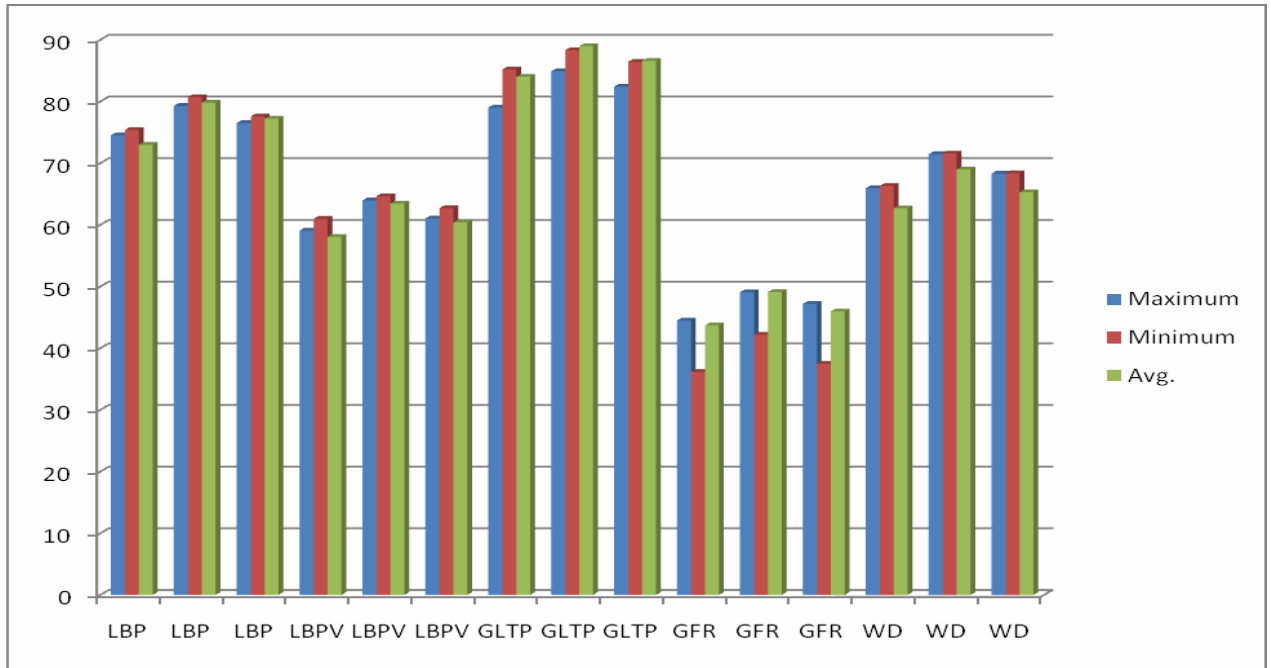
In this paper, we presented a novel model based on texture features for classification of kannada characters. We made a successful attempt to explore the applicability of texture features and wrapper feature selection methods for effective classification of character classification. In order to investigate the effectiveness and robustness of the proposed model, we conducted series of experiments on our own large dataset.

VII. REFERENCES

1. Hanmandlu M, A V Nath, A.C Mishra and V.K Madasu, Fuzzy Model Based



Figure 4.



Classification accuracy of the proposed model for classification based on individual texture models in experiment 1 (30% training sample)

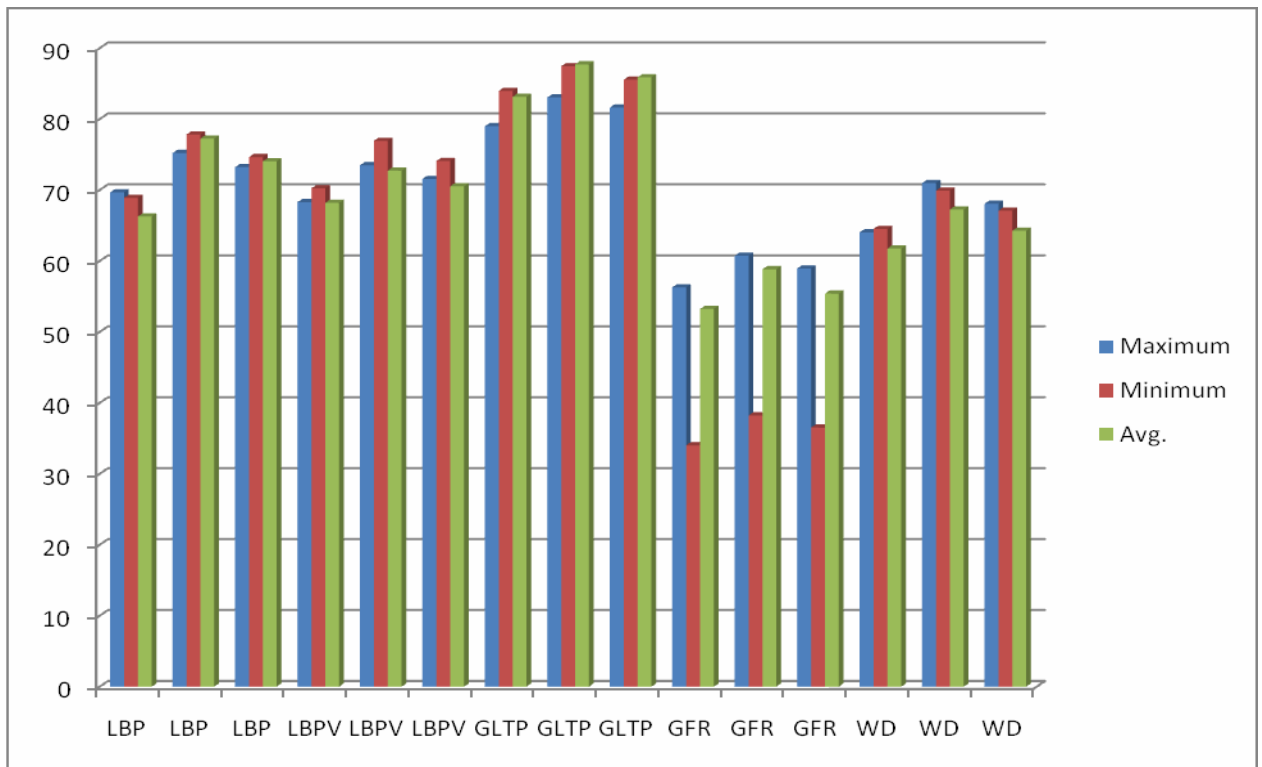


Figure 5. Classification accuracy of the proposed model for classification based on individual texture models in experiment 2 (40% training samples)

Textural Features and Feature Selection in KAN-NADA Character Recognition

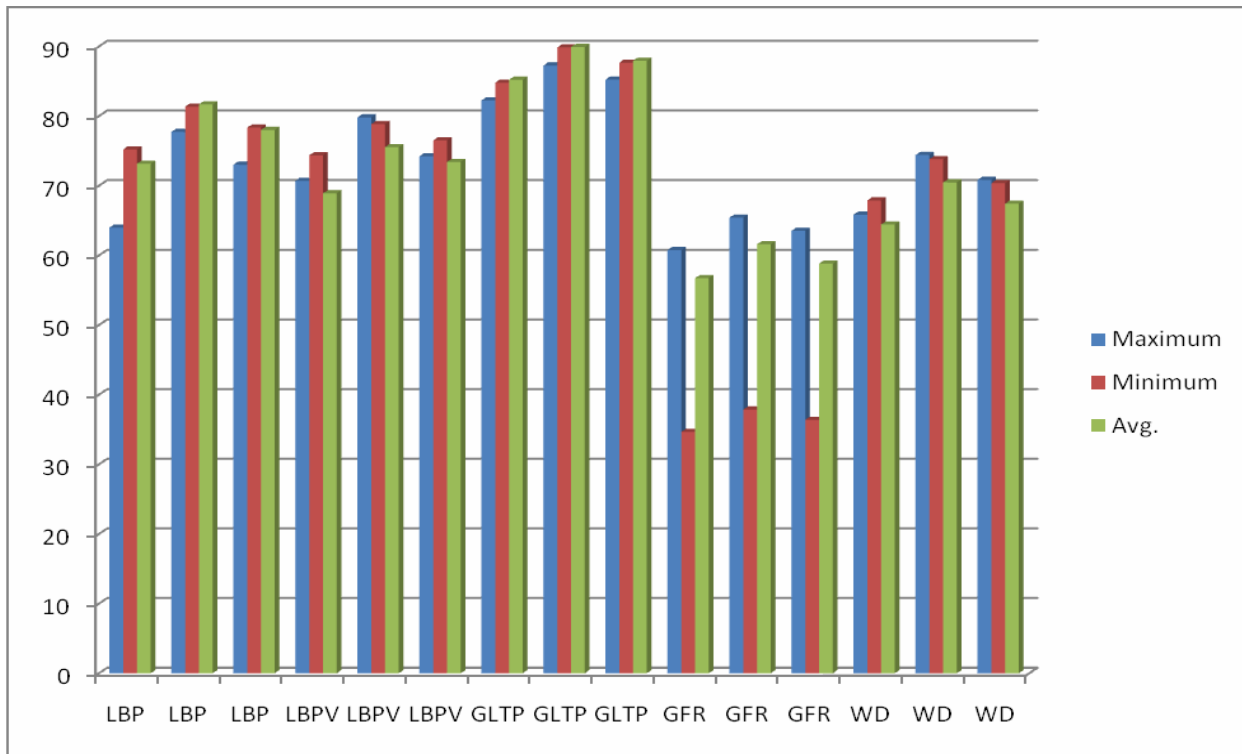


Figure 6. Classification accuracy of the proposed model for classification based on individual texture models in experiment 3 (50% training samples)

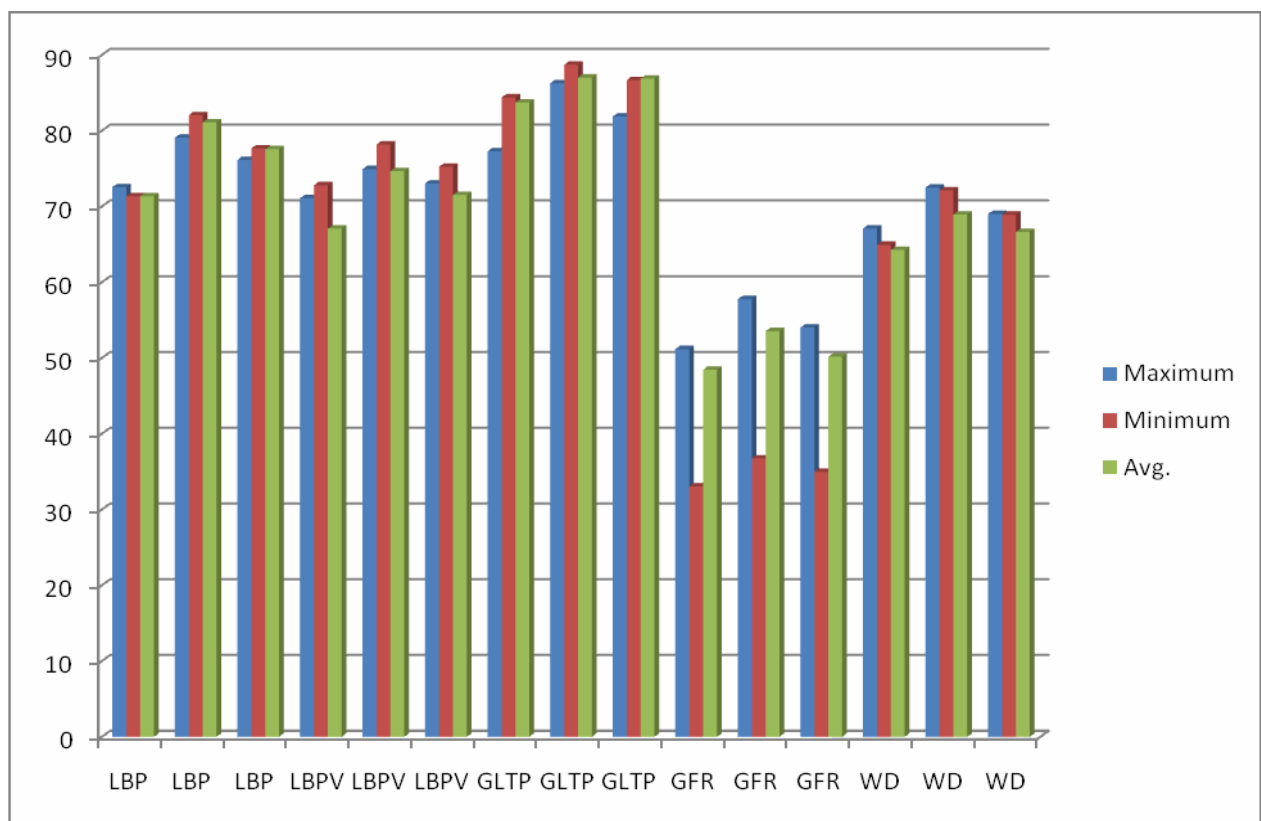


Figure 7. Classification accuracy of the proposed model for classification based on individual texture models in experiment 4 (60% training samples)

Table 2. Classification accuracy of the proposed model based on fusion of all five texture models

Texture features	Similarity Measures	Minimum accuracy	Maximum accuracy	Avg. accuracy
GLTP + LBP + LBPV + GFR + WD	Euclidean Distance	51.67	56.16	53.9
	Chisquare distance	53.39	58.65	56.20
	G-statistic distance	58.62	63.13	60.75

Table 3: Classification accuracy of the proposed model based on fusion of two texture models at a time and feature selection method

Texture features	Feature selection methods			
	SFS	SFFS	SBS	SFBS
GLTP	88.034	87.415	88.023	87.9115
LBPV	84.591	83.292	84.883	83.9350
GFR	87.540	87.163	88.652	86.5317
WD	86.965	87.408	90.434	86.0550
LBP	78.413	78.539	79.331	79.0528

Table 4. Classification accuracy of the proposed model based on fusion of all five texture models at a time and feature selection method

Texture features	Feature selection methods			
	SFS	SFFS	SBS	SFBS
GLTP + LBP + LBPV + GFR + WD	70.84	69.42	71.38	68.54

