Machine Reading of Arabic Manuscripts using KNN and SVM Classifiers

Aasim Zafar
Department of Computer Science
Aligarh Muslim University
Aligarh, INDIA
aasimzafar@gmail.com, azafar.cs@amu.ac.in

Arshad Iqbal
K. A. Nizami Centre for Quranic Studies
Aligarh Muslim University
Aligarh, INDIA
iqbal.arshadcqs@gmail.com

***Abstract*—In this paper, feature extraction techniques like Histogram of Oriented Gradients (HOG) and Local Binary Pattern (LBP) have been applied for feature extraction based on the structure of the Arabic Manuscript texts. Further, the two most popular classifiers namely, K-Nearest Neighbor (KNN) and Support Vector Machine (SVM) have been used to classify these texts and the results of both the classifiers have been compared. For comparison, we have used 1155 images of Arabic words and 1100 Kufic words for the purpose of training and testing and then the results of both the classifiers have been compared. In this work, we have used the Arabic *AHDB dataset* and *KUFIC dataset* for experimentation and for selecting, training and testing the data; the partition approach has been used. It is found that SVM performs better than KNN and achieved maximum recognition accuracy of 97.05% (with KUFIC dataset) and 97.80% (with AHDB dataset).**

Keywords—Arabic manuscript text; feature extraction; HOG;LBP;SVM; KNN.

I. INTRODUCTION

The main difficulty in Arabic handwritten text recognition is due to the cursive nature of Arabic script. In recent years a lot of researches have been carried out for recognizing Arabic script characters which undertook basic approaches of baseline detection and segmentation. Yet, these are not efficient as segmentation introduces undesirable noise and which poses challenges in recombining the characters. Also, various classifiers do not work well with segmented input.

Many offline Arabic handwritten recognition systems have been proposed to transliterate Arabic manuscripts into machine readable form. A large number of problems occur due to overlaps, touching words, text-line inclination, ligatures, irregular spaces between words, words without dots, etc. [1]. The diagrammatic representations (Fig. 1 and Fig. 2) highlight the problems mentioned above.

Fig. 1. Examples of ligatureword 'Kama'

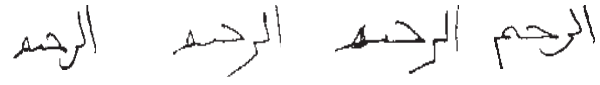


Fig. 2. Example of overlap word and word without dot'Ar-Rahim'

To develop any handwritten recognition system, a large number of sample patterns are required. There are many offline character/word databases such as AHDB [2], KHATT[3],IFN/ENIT[4], etc. available for offline handwritten Arabic recognition system. In this work, AHDB dataset has been used which is essentially meant for recognizing Arabic Handwriting. We have also used KUFIC dataset in this work for recognizing ancient Kufic Quranic manuscripts. This primary dataset has been prepared mainly with the help of [5].For this experimentation, we have used 1100 Kufic word images from this dataset, while 1155 word images from AHDB dataset. Authors in [6] have applied K-Nearest Neighbors classification for recognition of offline handwritten Arabic words. The recognition accuracy was 76.04%. In [7], authors have presented DCT and HOG techniques for feature extraction methods for recognizing Arabic words. The SVM classifier has been used for the recognition of Arabic words. The recognition rate was 96.31%. In [8], the authorshave reviewed many deep learning approaches for Arabic Handwriting Recognition. They also identified challenges and gave recommendations.Authors in [9] haveintroduced a new hybrid system for recognition of offline Arabic handwriting. This system was based on the combination of a neural network type multi-layer perceptron (MLP) and hidden Markov models (HMM).In [10],a robust system is presented for recognition of printed and isolated Arabic characters using the Hough Transform. This system was tested over 10 fonts and not affected by the noise and scaling.

The proposed work attempts to overcome some of the problems discussed above like the problem of overlap, irregular spacing, words without dots, ligatures, etc.To achieve these goals, we have proposed a system for Arabic manuscript text recognition by the way of preprocessing, segmenting words, extracting structural features from word images and recognizing by utilizing SVM and KNN classifiers.

The rest of the paper is organized as follows: The methodological steps and various tools and techniques used in the proposed system are presented in section II. Section III discusses the experimental results. The final conclusion is drawn in section IV.

II. PROPOSED SYSTEM

Three phases namely, preprocessing& segmentation, and feature extraction & classification are used for recognizing

978-93-80544-38-0 /20/$31.00 ©2020 IEEE

Arabic and Kufic texts. The diagrammatic representation of the proposed system is given in Fig.3.

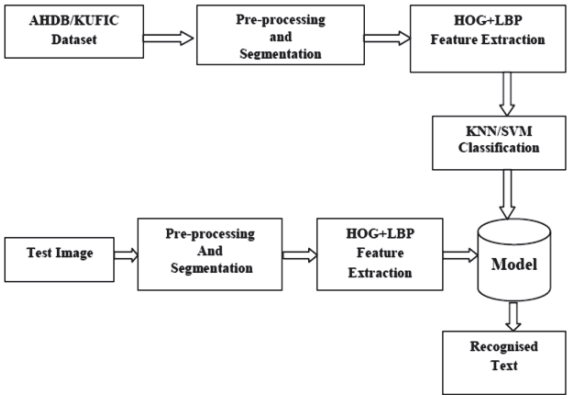
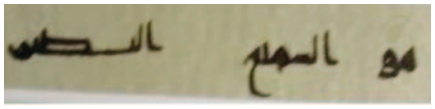


Fig. 3. Arabic manuscript recognition system

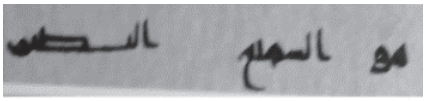
A. Preprocessing and Segmentation

Pre-processing involves various phases which enhance the accuracy of the proposed system. It is applied for both AHDB dataset and KUFIC image document. First, the Kufic image document is converted tograyscale Kufic script then *Wiener* filter has been used for removing the existing noise. During the pre-processing phase, this Kufic script gets converted to a binary image using a threshold function. This aids in the easy processing of the image. For thresholding purpose, we have employed adaptive thresholding. This introduces some noise in the image which is later on removed by using the *Median filter*. Then the Kufic image document is segmented into word level.

The last step towards achieving the goal of the pre-processing phase is to normalize the segmented images so that the segmented images are of the same size resulting in fast recognition.The images are then resized by 50x100 pixels.These images are used inthe feature extraction phase.The steps of this phase are shown in Fig. 4.



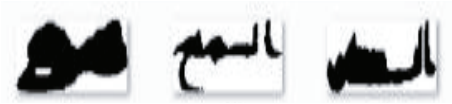
(a)



(b)

(c)

(d)



(e)

Fig. 4. Preprocessing andSegmentation(a) RGBImage, (b) Grayscale and Wiener Filtered Image, (c) ThresholdImage, (d) Median FilteredImage, and (e) SegmentedImage

B. Features Extraction

The most important phase in recognition of Arabic manuscript text is feature extraction. There are two broad categories of features; a structural feature that contains an intuitive aspect of writing like dot, end-point, loops, etc., and the statistical feature that contains numerical measures which are computed over regions of images. These include histograms, pixel densities, moments, etc. In this phase, a manuscript textimage is analyzed and a set of features are picked. These features are used for classifying the textsuniquely. The performance of a recognition system greatly depends on the extracted features.Two feature extraction approaches, namely, HOG [11] and LBP [12], which are commonly used texture descriptors, are discussed below.

C. Histogram of Oriented Gradient (HOG)

HOG is used as a very popular descriptor in pattern recognition. The steps used in generating HOG feature vectors are as follows.

- First, divide the image into the cells of 2x2 i.e. one block and a histogram of gradient directions for the pixels within the cell is computed for each cell.
- To find the gradient of an image, Eq. (1) is used as an edge detection mask filter in both the x-axis and y-axis.

$$\mathrm{Dx} = [-1\ 0\ 1] \text{ and } \mathrm{Dy} = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \quad (1)$$

- Using a convolution operation, obtain Gradient in x and y directions of an image I usingEq. (2).

  Gradient in x-direction (g_x)=I * D_x and

  Gradient in y-direction (g_y)=I* D_y (2)

  Now,obtain the magnitude of gradient usingEq. (3) and obtain orientation of the gradient using Eq. (4).

$$g = \sqrt{g_x^2 + g_y^2} \quad (3)$$

$$\theta = \tan^{-1}\left(\frac{g_x}{g_y}\right) \quad (4)$$

- 9 bins consist from 0 to 180 degrees for the histogram channel.
- The pixel of each cell contributes a weighted gradient to the corresponding bin.

- Normalize the group of histograms of one block
- Once the block normalization is done, the histograms of all blocks are gathered into one big HOG feature vector.

Fig.5 shows the histogram of HOG and Fig.6presents application of HOG descriptor of Kufic word 'As-Sami'.

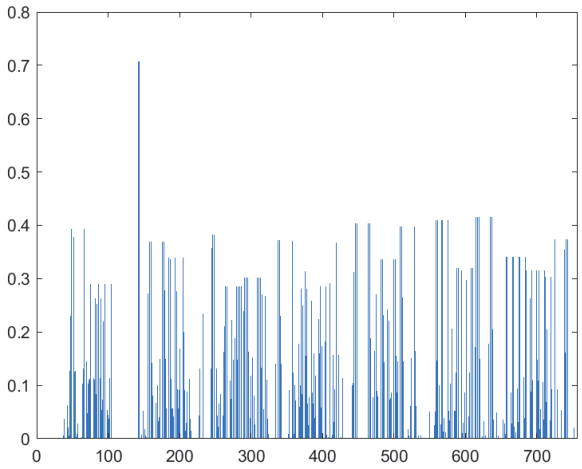


Fig. 5. HOG histogramof 'As-Sami'

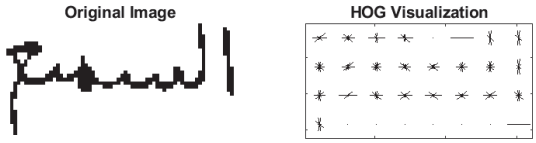


Fig. 6. HOG visualization of 'As-Sami'

D. Local Binary Pattern (LBP)

The steps used in generating the LBP feature vector are as follows:

- Split the window into cells (each cell of 8x8 pixels).
- Compare each pixel of a cell to each of its 8 neighbours in clockwise or anti-clockwise.
- Depending on the value of the surrounding pixel it is set to 0 (if value < central pixel value) otherwise 1(if value >= central pixel value). This produces a binary number of 8-digit, which is then converted to a decimal number.
- Using eight neighbouring pixels, the final value will be between 0 and 255.This is called 256-Level Encoding LBP.

  256-Level Encoding is ineffective in the case of noisy regions and overfitting inthe case of low-resolution images [13]. Non-Uniform LBP is used to remove LBPredundancyfrom noisy regions. Non-UniformLBP is a bit sequence with more than 2 bitwise transitions (0->1 or 1->0).

  Then, we convert 256-Level Encoding to 59-Level Encoding

  In 59-Level Encoding, there are 58 uniformsLBP, assign each with a unique index (from 0 to 57) and 198 non-uniforms LBP, assign all with index 58.
- Concatenate the normalized histogram of all cells.This gives feature vectors of 59.

Fig.7 shows the histogram of LBP and Fig.8represents application of LBP descriptorof Kuficword 'As-Sami'.

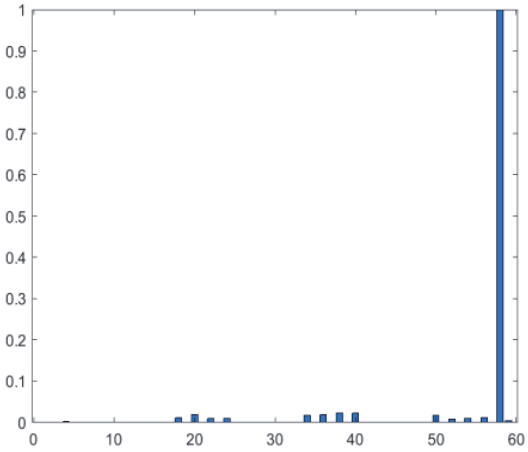


Fig. 7. LBP histogram of 'As-Sami'

Fig. 8. LBP visualization of 'As-Sami'

E. Classification

The outputs of HOG and LBP feature vectors are concatenated to form resulting feature vectors for classification. In this proposed system, KNN and SVM classifiers are used for text recognition.

F. KNN Classifier

KNN algorithm is a famous classification techniquethat assigns an object or a test pattern to a class based on the majority of its K-nearest neighbors in the feature space.The process of classifying any object by KNN is easy. The classifier uses training sets with class labels. To assign a class for a test sample, first,calculate its distance to each training sample. The distance is computed by Euclidean distance, as given in Eq. (5).

$$d(x,y)=\sqrt{\sum_{i=1}^{m}(x_i-y_i)^2} \qquad (5)$$

Where m=features in the dataset and x_i & y_i are values of features i in objects x and y respectively. If k=1, then it is the simplest case of KNN.In Fig. 9,the black star (test data) belongs

to Class B, because the black star is close to 2 green stars of class B than 1 red star of class A,k=3.

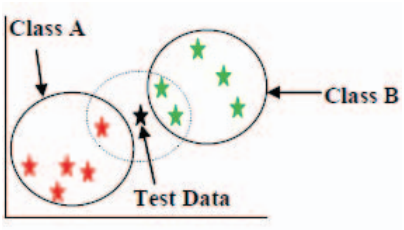


Fig. 9. Test data(black star) belongs to Class B

G. SVM Classifier

SVM is a supervised learning technique, which was first introduced by [14] and used for regression and classification problems. SVM is used as a binary linear classifier as well as non-linear or multiple class problems[15].

The main objective of SVM is to draw an optimal hyperplane as a decision boundary in such a manner that maximized the margin of two classes (+ve and –ve).Fig. 10depicts the working of SVM.Eq. (6) represents the equation of optimal hyper plane.

$$wx + b = 0 \qquad (6)$$

where w is weight vector and b is a bias.

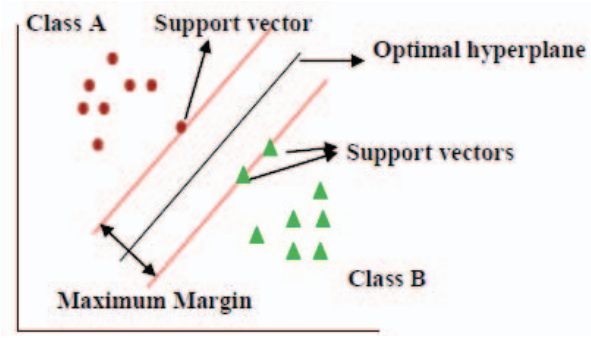


Fig. 10. Support vector machine

Linear, RBF and polynomial kernels are used for multiclass SVM classification.

III. Experiment and Results

The results of a recognition system for Arabic and Kufic manuscripts text are presented in this section. We have divided both the datasets (AHDB and KUFIC) using five partitioning approaches, namely x1, x2, x3, x4 and x5 as per the description given in Table I.

TABLE I. Partition Detail of Dataset

Approach	Partition Detail	
	Training Set in (%)	*Test Set in (%)*
x1	50	50
x2	60	40
x3	70	30
x4	80	20
x5	90	10

The extracted features of HOG and LBP are applied to two classifiers, namely, KNN and SVM for recognition. Three kernels namely, linear kernel, RBF kernel, and polynomial kernel are used in SVM. The results are given in Table II. Interpretation of the experimental results for both the datasets are presented below.

TABLE II. Comparison of Recognition Accuracy between KNN and SVM Classifiers

Classifier	Partition (0.5,0.5) *% Accuracy*	Partition (0.6,0.4) *% Accuracy*	Partition (0.7,0.3) *% Accuracy*	Partition (0.8,0.2) *% Accuracy*	Partition (0.9,0.1) *% Accuracy*	Data Set
KNN	93.27	93.48	94.03	94.20	**94.55**	
SVM						
linear	95.09	95.15	94.94	95.00	94.75	KUF
RBF	95.27	96.36	96.49	96.25	96.26	IC
polynomial	96.55	96.21	96.62	**97.05**	96.97	
KNN	93.65	94.52	93.73	**94.59**	94.26	
SVM						
linear	94.68	94.95	94.47	94.70	95.69	AH
RBF	96.05	96.68	97.30	96.86	**97.80**	DB
polynomial	96.40	96.25	97.17	97.08	97.32	

A. Recognition accuracy using KNN vs SVM kernels in KUFIC dataset

The experimental recognition results for all five partitioning approaches (x1, x2, x3, x4, and x5) based on the KUFIC dataset using KNN and SVM with linear, RBF, polynomial kernels are presented.We have achieved a maximum accuracy of 94.55% using KNN when we used strategy x5 and achieved an accuracy of 97.05% using SVM with the polynomial kernel when we used the strategy x4 in KUFIC dataset. These results are shown in Fig.11.

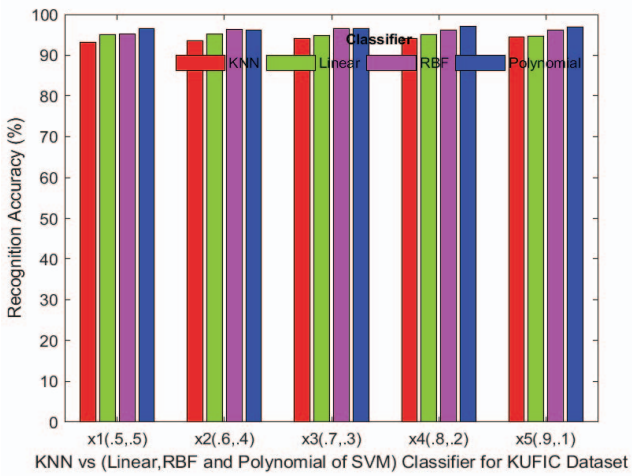


Fig. 11. Recognition accuracy using KNN vs SVM kernels in KUFIC dataset.

B. *Recognition accuracy using KNN vs SVM kernels in AHDB dataset*

With AHDB dataset, the recognition results for KNN and SVM kernels have been comparedfor all partitioning approaches. We have achieved a maximum accuracy of 94.59% using KNN with strategy x4 and achieved an accuracy of 97.80% using SVM with RBF kernel when we used the strategy x5 in AHDB dataset. These results are shown in Fig. 12.

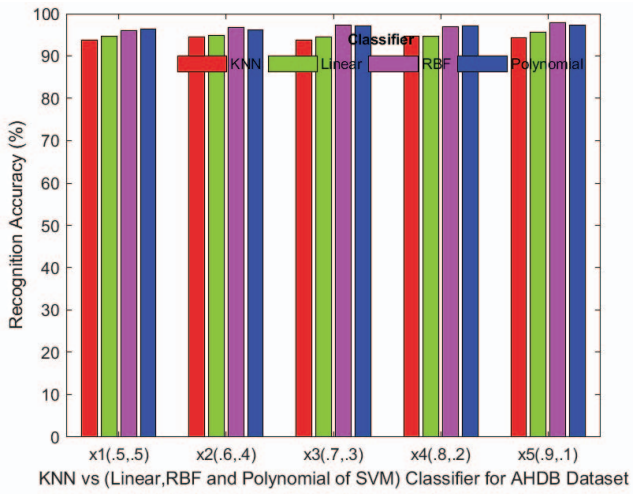


Fig. 12. Recognition accuracy using KNN vs SVM kernels in AHDB dataset

IV. Conclusion

A comparison of KNN and SVM classifiers to recognize Arabic manuscripts texts has been presented. The features of a text have been extracted using HOG and LBP descriptors. For comparison of recognition accuracy, KNN and SVM with three types of kernels, i.e., linear, RBF and polynomial have been used on AHDB dataset and KUFIC dataset separately. The concatenated features of HOG and LBP have been input to both classifiers. We achieved a maximum accuracy of 94.55% using KNN when we used strategy x5 and achieved an accuracy of 97.05% using SVM with polynomial kernel when we used the strategy x4 in KUFIC dataset whereas we achieved a maximum accuracy of 94.59% using KNN when we used strategy x4 and achieved an accuracy of 97.80% using SVM with RBF kernel when we used the strategy x5 in AHDB dataset. The results clearly indicate that SVM performs better than KNN for both the datasets. Further, the encouraging results inspire that this work can also be extended for other manuscripts other than Kufic script.

References

[1] N. Aouadi and A. K. Echi, "Word Extraction and Recognition in Arabic HandwrittenText," *International Journal of Computing & Information Sciences*,vol. 12, no.1, pp. 17-23, 2016.

[2] S. Al-Ma'adeed, D. Elliman, and C.A. Higgins, "A database for Arabic handwritten text recognition research," in *Proc. of the 8th International Workshop on Frontiers in Handwriting Recognition, Canada*, 2002.

[3] S. A. Mahmoud, I. Ahmad, M. Alshayeb, W. G. Al-Khatib, M. T. Parvez, G. A. Fink, V. Margner, andH. E. Abed, "KHATT: Arabic Offline Handwritten Text Database," in *Proc. of the International Conference on Frontiers in Handwriting Recognition (ICFHR 2012), 2002*, pp. 449-454.

[4] M. Pechwitz, S. S.Maddouri, V. Märgner, N. Ellouze, and H.Amiri, "IFN/ENIT- Database of Handwritten Arabic Words," in *Proc. of Francophone International Conference on Writing and Document,2002*, pp. 127-136.

[5] M. M. Al-Azami, *Ageless Qur'an Timeless Text*. Azami Publishing House, 2017.

[6] J.H. Alkhateeb, F. Khelifi, J. Jiang, and S. S. Ipson, "A new approach for off-line handwritten Arabic word recognition using KNN classifier,"in *Proc. of the IEEE International Conference on Signal and Image Processing*, 2009, pp. 191-194.

[7] A. K. A. Hassan and M. S. Kadhm, "Arabic Handwriting Text Recognition Based on Efficient Segmentation, DCT, and HOG Features," *International Journal of Multimedia and Ubiquitous Engineering*, vol.11, no. 10, pp.83-92, 2016.

[8] A. S. Ahmed, S. Awang, W. Al-Saiagh, S. Tiun, and A. S. Al-Khaleefa, "Deep learning algorithms for arabic handwriting recognition: A review," *International Journal of Engineering and Technology (UAE)*, vol. *7,* pp. 344-353, 2018.

[9] K. Fergani and A. Bennia,"New Segmentation Method for Analytical Recognition of Arabic Handwriting using a Neural-Markovian Method," *International Journal of Engineering and Technologies*, vol. 14, pp. 14-30, 2018.

[10] M. Kadi, "Isolated Arabic Characters Recognition, Using a Robust Method Against Translation, Noise, and Scaling, Based on the Hough Transform," *International Journal of Information Science & Technology*, vol.3, no.4,pp.34-43, 2019.

[11] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection,"in *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 2005*, pp. 886-893.

[12] T. Ojala, M. Pietikainen, and D. Harwood, "Performance evaluation of texture measures with classification based on kullback discrimination of distributions,"in *Proc. of the International Conference on Pattern Recognition*, 1994.

[13] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution Gray Scale and Rotation Invariant Texture Classification with Local Binary Patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971-987,2002.

[14] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers,"in *Proc. of the Annual Workshop on Computational Learning Theory*, 1992, pp. 144-152.

[15] C. Campbell and Y. Ying, "Learning with support vector machines,"*Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 5, no. 1, pp.1-95, 2011.