Numeral Script Identification from Handwritten Document Images

Article *in* Procedia Computer Science · August 2015

DOI: 10.1016/j.procs.2015.06.067

4 authors:

Obaidullah Md Sk
Aliah University
116 PUBLICATIONS **678** CITATIONS
SEE PROFILE

Chayan Halder
Ramakrishna Mission Vivekananda Centenary College
46 PUBLICATIONS **365** CITATIONS
SEE PROFILE

Nibaran Das
Jadavpur University
197 PUBLICATIONS **2,592** CITATIONS
SEE PROFILE

Kaushik Roy
West Bengal State University
251 PUBLICATIONS **1,803** CITATIONS
SEE PROFILE

Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 54 (2015) 585 – 594

Procedia
Computer Science

Eleventh International Multi-Conference on Information Processing-2015 (IMCIP-2015)

Numeral Script Identification from Handwritten Document Images

Sk Md Obaidullah[a,*], Chayan Halder[b], Nibaran Das[c] and Kaushik Roy[b]

[a]*Department of Computer Science & Engineering, Aliah University, Kolkata, 700 156, India*
[b]*Department of Computer Science, West Bengal State University, Kolkata, 700 126, India*
[c]*Department of Computer Science & Engineering, Jadavpur University, Kolkata, 700 032, India*

Abstract

In this paper a novel *HNSI* (Handwritten Numeral Script Identification) framework to identify scripts from document images containing numeral text written by any one of the four popular Indic scripts namely Bangla, Devanagari, Roman and Urdu has been proposed. A dataset of 4000 word-level numeral images with equal distribution of each script type are collected from different individuals with varying age, sex and educational qualification. Some spatial and frequency domain features has been computed and a 55-dimensional feature vector is developed. During experimentation the whole dataset is divided into 2:1 ratio for training and testing. Performance of different classifiers is compared and MLP is found to be the best one while evaluating accuracy rate for combinations like Four-scripts, Tri-scripts and Bi-scripts.

Keywords: Benchmark result; Daubechies wavelet; MLP classifier; Multi-script handwritten document image; Numeral script identification.

1. Introduction

India is a Sub-continent where 23 different languages (including English, which is very popular here) are present and 13 different scripts (including Roman) are used to write them[1,2]. Here an official document can be written using multiple scripts creating a multi-script document. To process those multi-script documents *OCR* (Optical Character Recognizer) is used as an intelligent technique which converts image to text version. But designing a single *OCR*[3] which will cater all the official scripts of India is really a challenging and infeasible task. That is why prior identification of script type is necessary before feeding the document image to the script specific *OCR*. Document image processing researchers are working towards this goal to develop different script identification techniques which will identify scripts from multi-script document images. The whole work of script identification can be divided into two categories namely *PSI* (Printed Script Identification) and *HSI* (Handwritten Script Identification) depending on the input acquired. Developing *HSI* technique are more challenging than *PSI* due to dynamic nature of handwritten data compared to printed one. Again each of *PSI* and *HSI* techniques can be categorized into Document-level, Block-level, Line-level, Word-level and Character-level based on presence of the multi-script texts in real life document images.

*Corresponding author. Tel.: +91-9830368169
E-mail address: sk.obaidullah@gmail.com

doi:10.1016/j.procs.2015.06.067

০	১	২	৩	৪	৫	৬	৭	৮	৯

Fig. 1. Standard Bangla numerals.

०	१	२	३	४	५	६	७	८	९

Fig. 2. Standard Devanagari numerals.

0	1	2	3	4	5	6	7	8	9

Fig. 3. Standard Roman numerals.

1.1 Motivation

All the works available in literature are mainly based on script identification on alphabetic characters. Till date very few works[4] has been reported on *HNSI* (Handwritten Numeral Script Identification) and no work considering four numeral Indic scripts, which inspired us to carry out the present work. *HNSI* has its applicability in different domain of 'smart computing' like automatic sorting of postal documents based on PIN code script, automatic classification of application forms, examination forms etc. written by native languages based on a numeral strings. The present work proposes an intelligent *NSI* technique from handwritten document images written by any one of the four popular Indic scripts namely *Bangla, Devanagari, Roman and Urdu*. We have also applied different feature combinations along with different script combinations to make the system more robust.

1.2 Properties of indic numeral scripts

1.2.1 Bangla

Bangla is one of the most popular scripts in Eastern India. Bangla numeral system is followed by Bangla, Assamese and Manipuri languages. Different eastern Indian states and county like Bangladesh use this numeral script. Figure 1 shows different numeral under this script.

1.2.2 Devanagari

Devanagari numeral system is followed by popular North Indian languages like Hindi, Sanskrit, Marathi etc. Four numeral digits of Devanagari script are almost structurally and visually similar with Bangla. Figure 2 shows different numeral under this script.

1.2.3 Roman

Roman is a popular script throughout the Indian Sub-continent which is used by English and Santhali language. Four numeral digits of Roman are almost structurally and visually similar with Bangla and Devanagari script. Figure 3 shows different numeral under this script.

1.2.4 Urdu

Urdu is a variant of Eastern-Arabic family of numerals. Urdu numeral script is very much structurally and visually distinct in comparison with Bangla, Devanagari and Roman. Only one Urdu numeral digit looks similar with other three. Figure 4 shows different numeral under this script.

Considerable number of works on Indic script identification has been carried out by the document image processing researchers. The *PSI* techniques are reported at[5–9] and *HSI* technique on Indic scripts are mentioned at[10–14]. But

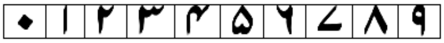

Fig. 4. Standard Urdu numerals.

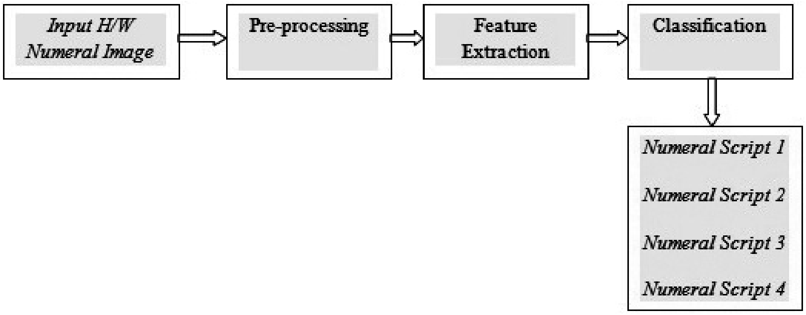

Fig. 5. Block diagram of the proposed handwritten NSI system.

all these *PSI* and *HSI* works has been carried out on alphabetic characters at different levels. But the efforts on *HNSI* techniques in literature are very less till date. U. Pal *et al.*[15] in a work on Indian postal automation proposed a scheme for multi-script PIN code string recognition based for different script type. Here they had considered Roman, Devanagari and Bangla script for their work. To the best of our knowledge the present work is the first of its kind on *HNSI* considering four popular Indic scripts namely Bangla, Devanagari, Roman and Urdu at Word-level.

A block diagram of the proposed system is shown in Fig. 5, where handwritten numeral images are supplied as inputs, and then pre-processing is done which includes segmentation and binarization. In next step suitable features are extracted from those pre-processed images. Finally classification is done and particular numeral script type by which the original document was written is produced as output.

The rest of the paper is presented as follows: Section 2 discuss about the proposed methodology where different subsections like segmentation, binarization, and feature extraction are available. Experimental details with data collection, evaluation protocol and experimental results are provided in section 3. Section 4 discuss about conclusion and future direction.

2. Proposed Work

Present work proposes a novel framework for handwritten numeral script identification. Due to unavailability of Word-level numeral images, the authors were devoted a considerable amount of time towards data collection. Afterwards pre-processing is done which includes operations like segmentation and binarization. Following section discuss about different pre-processing techniques.

2.1 Preprocessing

2.1.1 Segmentation

Segmentation is a crucial task for the current work. The data collection is done in a standard pre-formatted A4 size page where different writers were asked to write an unconstrained numeral word consisting of three to five digits. Detail about data collection procedure is provided in section 3. Initially collected documents are digitized using a HP flatbed scanner at 300 dpi. For segmenting the numeral word from the original page a line structuring element has been created whose dimension was fixed experimentally and morphological dilation is done on the complemented version of the threshold image to create single block for each of the word image. Then component labeling is done and numeral word blocks are extracted applying bounding box technique on the original image. A threshold block area size (min 30000, max 60000) was heuristically determined earlier. Measures were taken to remove unwanted pixels

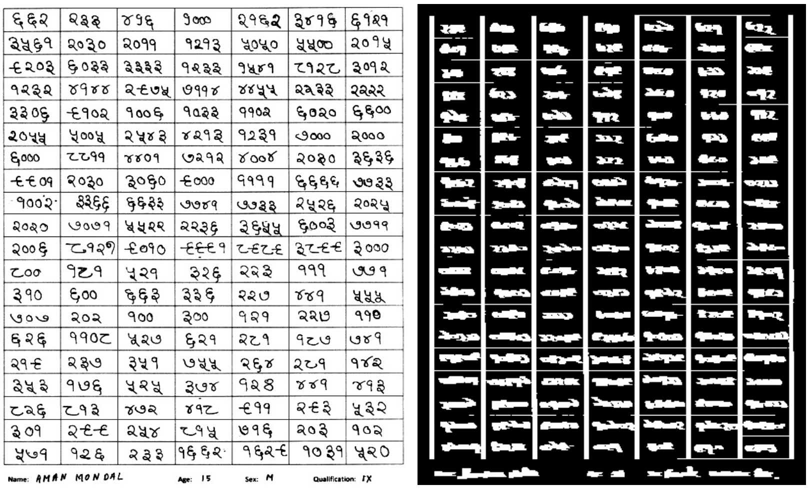

Fig. 6. (a) Original numeral data collection form of Devanagari script; (b) Numeral word block segments.

at boundary region and finally using a bicubic interpolation technique all numeral images are resized at dimension of 140 × 340 pixels and stored in .jpg format. Figure 6a shows a sample data collection form and an intermediate phase of segmentation where numeral word blocks are created (Fig. 6b).

2.1.2 Binarization

After extracting 140 × 340 numeral word segments they are converted into binary version by applying a two-stage based binarization technique[16]. Pre-binarization is done at first stage based on local window algorithm in order to get an idea of different region of interest. *RLSA* (Run Length Smoothing Approach) is applied on the pre-binarized image afterwards, to reduce the hollow/stray regions produced due to the local binarization method used earlier. Then, component labeling is done, where each component is selected and mapped in the original gray image to get respective zones of the original image. The final 0-1 image is obtained by applying a global binarization algorithm on these regions of the original image. The advantage of using two-stage algorithm is the output will be at least as good as if only global binarization method would have been used.

2.2 Feature extraction

Feature extraction is one of the most important tasks in any pattern recognition work. Selection of ‘suitable’ feature set leads to better performance in terms of final accuracy of the system. Here ‘suitable’ means those set of features which will be capable enough to capture maximum inter-script variation still easily computable. We have calculated both frequency and spatial domain features for this work. Initially frequency domain features namely daubechies wavelet decomposition is done at level one for different spectral coefficients. Then different spatial domain features namely entropy, mean, standard deviation etc. are calculated. Finally a 55 dimensional feature set is constructed. Following section describe briefly about features extracted from the numeral word image of different scripts.

2.2.1 Frequency domain feature

Any image can be processed either in spatial or frequency domain. Spatial domain processing of an image includes processing of its pixel values as it is. Whereas the rate of changes of different pixel values of an image are analyzed in frequency domain. During frequency domain analysis different variations of wavelets are used successfully in many image processing applications[17]. The present work is based on *DWT* (Discrete Wavelet Transform)[20] where different wavelets are discretely sampled. Wavelet has its advantage over popularly used Fourier transform in terms of temporary

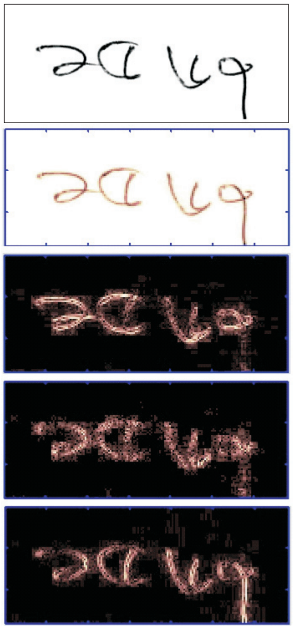

Fig. 7. Computation of different Daubechies wavelet coefficients at level 1 on Bangla numeral Word-level image (top to bottom: original Bangla word image, approximation coefficient *cA1*, horizontal coefficient *cH1*, diagonal coefficient *cD1*, vertical coefficient *cV1*.

resolution, which means it can capture both frequency and location information with respect to time. There are many popular wavelet transform techniques available till date, out of which we have used Daubechies wavelet families for the present work.

Daubechies wavelet transform

Daubechies wavelets belong to the family of discrete wavelet techniques which is based on the idea of Ingrid Daubechies. Their advantages include easy computation with minimal resource and time requirement. These orthogonal wavelets are characterized by maximum number of vanishing moments for some given support. A signal (here an image) can be decomposed into different frequencies with different resolution using wavelet decomposition. That is why wavelets are used for multi-resolution analysis. In general Daubechies wavelet is denoted as '*dbN*', where '*db*' denotes wavelet family and '*N*' represents the number of vanishing moments. The polynomial behavior of an image is represented by its number of vanishing moments. For example the constant coefficient component of an image signal can be represented by Daubechies single moment wavelet decomposition or *db1*. Similarly *db2* and *db3* can represent components of an image signal with linear and quadratic coefficients. In present work Daubechies wavelets for *db1, db2 and db3* are computed on the original numeral Word-level images generating four coefficients namely approximation coefficients (*cA*), horizontal coefficients (*cH*), vertical coefficients (*cV*), and diagonal coefficients (*cD*). Then different spatial properties namely entropy, mean, standard deviation etc. are computed on each of these coefficient matrices. Figure 7 shows original Bangla numeral Word-level image and its different sub-band images computed at level 1.

2.2.2 Spatial domain feature

Frequency domain analysis is the background of representation of the feature vector. Different textural and statistical values are also computed which enriches our present feature vector. Entropy, which is a statistical measurement of the randomness of an image, is computed on grey-scale, binary, and twelve sub-band images (for *db1, db2 and db3*). Similarly, statistical measures namely mean, standard deviation are also computed on the same set of images to enrich our feature vector.

Based on the above mention techniques a 55 dimensional feature vector has been constructed by the following algorithm:

Proposed algorithm

Input: Numeral Word-level grey scale image
Output: 55 dimensional feature set [F_1 to F_{55}]

Step 1: Automatically segmented Word-level images are read from the specified folder and converted into 0-1 level images using our existing two-stage based binarization algorithm.

Step 2: Texture analysis on gray and binary image is done applying statistical features like entropy, mean, standard deviation. This step will generate four features [F_1 to F_4].

Step 3: Daubecheis wavelet decomposition for constant coefficient of the source images (*db1*) is performed at level 1. Approximation coefficient *cA1*, horizontal coefficient *cH1*, diagonal coefficient *cD1*, vertical coefficient *cV1* are also computed. Entropy, mean and standard deviation are computed on each of the decomposed sub-images generating twelve features [F_5 to F_{16}].

Step 4: Wavelet entropy using five entropy types namely 'shannon', 'log energy', 'threshold', 'sure', 'norm' are computed on approximation coefficient *cA1* sub-image. This step will generate five features [F_{17} to F_{21}].

Step 5: Step 3 and 4 are repeated for linear and quadratic coefficients of the source images (*db2* & *db3*) generating 17 features for each [*db2:* F_{22} to F_{38} & *db3:* F_{39} to F_{55}].

Repeat *Step 1 to 5* for each of the 1000 word images of four scripts and finally a feature vector of 55 dimensions is generated.

3. Experimental Details

Experimental phase consists of development of dataset, choosing appropriate evaluation protocol, classification and finally analysis of the results. Following section discuss about all these tasks carried out during experimentation.

3.1 Dataset development

Data collection is the most imperative and time consuming task for the present work. Few numeral dataset is available which could not solve our purpose because they were developed mainly for *OCR*[18]. For script recognition purpose at Word-level we were compelled to prepare our own dataset. Initially a standard data collection format was prepared by creating a grid of 20 rows and 7 columns. Enough space was there in each box in the grid to write a three to five digit numeral word for common writers. Then these forms were given to different writers with varying age, sex and educational qualification. Here we have followed the Word-level approach because we felt that if writers were asked to write isolated numeral then they may write more carefully creating the loss of realness. That's why they were asked to write each numeral word of length three to five digits so that maximum realness within the data can be achieved as well as some real life applications data can also be modelled. Then applying segmentation technique mentioned in section 2, numeral word blocks of size 140 × 340 were generated. Total 4000 numeral word image blocks of equal distribution of each script type were prepared. Figure 6 shows one of the sample data collection forms of Bangla numeral script. Figure 8 shows sample numeral word images of Bangla, Devanagari, Roman and Urdu scripts.

3.2 Evaluation protocol

This section includes training and testing based on the extracted feature set. The learning process of distinguishable properties for each target script class is performed during training phase of classification. Whereas the test phase initiates the measurement of dissimilarity of the script classes based on the knowledge acquired during training process. For present experiment the whole dataset is divided into 2:1 ratio as training and test data. Following section describes about the outcome of the test phase.

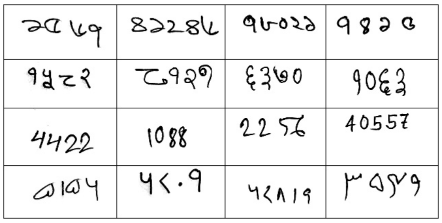

Fig. 8. Sample unconstrained 140 × 340 handwritten numeral block images of Bangla, Devanagari, Roman and Urdu scripts (top to bottom) automatically extracted.

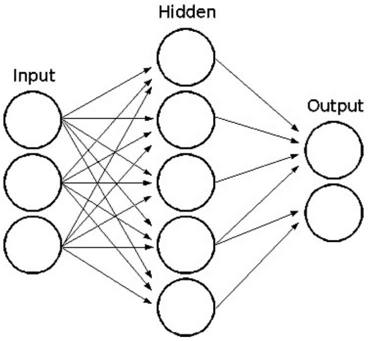

Fig. 9. A sample MLP with input, hidden and output layer. For present experiment the designed MLP is of 55-15-4.

3.3 Classifier

Seven different classifiers namely NBTree, PART, LIBLinear, Random Forest, SMO, Simple Logistic and MLP are experimented during classification[19]. Their performance is evaluated with respect to TP Rate (True Positive). MLP is found to be the best one among all in terms of AAR (Average Accuracy rate). Multi layer perceptron is a type of feed forward *ANN* (Artificial Neural Network) which is nothing but a mapping from an input set to output set. It can be represented by a direct acyclic graph where direction of the signal flow is specified. Each node of a *MLP* is mimicking of an artificial neuron. In the connection between two neurons a weight or label is associated which represents the capacity or strength of the connection. The number of neurons in input layer is same as the number of feature selected for the particular pattern recognition problem. Whereas the number of output layer is same as the number of target classes. Figure 9 shows a sample MLP classifier having input, hidden and output layers. The sum is then transformed using a transfer function. The strength of the synaptic connections are tuned during the training process so that the *MLP* can respond properly to every input values taken from the training set. For present work numbers of neurons in input layer are fifty five and neurons in output layers are four. The total number of hidden layers is computed empirically and the number of neurons present in each hidden layer should be determined during training process by trial run[16,19].

For the present work, size of the feature vector is 55 and the number of output class in 4, so the number of neurons in input layer and output layer are 55 and 4 respectively. The number of neurons for the hidden layer is calculated as 15 experimentally.

3.4 Result & analysis

HNSI from document images is a challenging task because number of samples in each script are limited to ten (zero to nine) and many samples are common to each other for different script pairs[4]. We have statically analyzed this similarity percentage from real life handwritten numeral data. It has been found that Bangla-Devnagari group has almost 40% numerals which are visually and structurally similar. So here, total number of numeral samples of these two scripts reduces to 16 instead of 20. This fact leads to higher misclassification rate between these two scripts. In a similar manner, Bangla-Roman, Devnagari-Roman and Roman-Urdu script pairs have digit similarity percentage of 30%, 40% and 20% respectively. It has been found that only Urdu script characters are almost distinct in visually and structurally from other three scripts. So whenever there is a pair of Urdu and any other script then reasonable outcome has been found. The effect of the similarity percentage among different scripts has direct impact on the identification rate. That is why numeral script identification from handwritten document images is a real challenge in terms of successful identification rate and still it is far from complete solution.

Extensive experimentation has been carried out for the present work. Performances of Bi-scripts, Tri-scripts, Four-scripts combinations are measured. Table 1 shows confusion matrix of Four-scripts on the test dataset. The last

Table 1. Confusion matrix on the test dataset after splitting the whole dataset into 2:1 training and testing set ratio.

Classified As	Bangla	Devanagari	Roman	Urdu
Bangla	222	62	23	13
Devanagari	91	218	18	30
Roman	65	36	142	91
Urdu	15	15	12	307
		Average Four-scripts identification Rate: 65.4%		

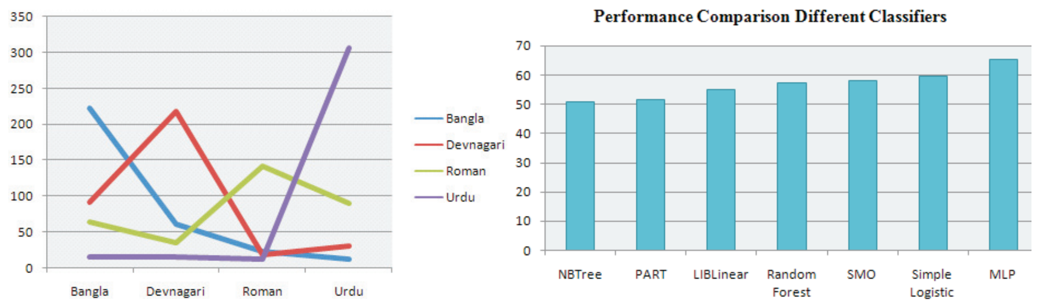

Fig. 10. (a) The graphical representation of the confusion matrix on the test dataset using MLP; (b) Performance comparison of seven different classifiers by average accuracy rate (%) measured using true positive values.

Table 2. Tri-script identification rate using MLP classifier on the test dataset of ${}^4C_3$ sets.

Sl. no.	Scripts combination	AAR (%)
1	{Bangla, Roman, Urdu}	74.6
2	{Devanagari, Roman, Urdu}	73
3	{Bangla, Devanagari, Urdu}	71.4
4	{Bangla, Devanagari, Roman}	68.2
5	**Avg. Tri-script Acc. Rate**	**71.8%**

row shows Four-scripts identification rate which is 65.4% for the present experiment. Maximum misclassification is found among Bangla, Devanagari and Roman scripts. This is due to the similar shaped digits of those scripts as mentioned in the introduction section. As per our expectation performance of Urdu script is encouraging among the four. This is because except the numeral 'one' in Urdu which is very much similar with numeral 'one' in Roman, all other digits are visually and structurally distinct in comparison with other three. The evidence of our observation can be seen from misclassification rate of Urdu script, which is almost equal with other three scripts (Bangla: 4.2%, Devanagari: 4.2% and Roman: 3.4%). Figure 10(a) shows graphical representation of Table 1, whereas the comparative graph of seven classifiers for Four-scripts average accuracy rate is shown by Fig. 10(b). The performances of Tri-scripts and Bi-scripts combinations can be found from Table 2 and Table 3 respectively. From Table 2 average Tri-scripts accuracy rate of 71.8% can be found for ${}^4C_3$ or four different combinations. Highest accuracy rate is reported by the {Bangla, Roman, Urdu} combination which is 2.8% more than Tri-scripts average rate. Whereas the Tri-script combination of {Bangla, Devanagari, Roman} reports the lowest accuracy rate among all which is 3.6% below the average rate. Among the ${}^4C_2$ or six Bi-scripts combinations highest accuracy is found for the script combination {Bangla, Urdu} (90.9%) combination, followed by {Devanagari, Urdu} (89.6%) and {Roman, Urdu} (82.1%). Lowest Bi-script accuracy rate is found for {Bangla, Devanagari} script combination which is 10.4% below the average Bi-scripts accuracy rate (82.2%). The pattern of the result is also similar here in comparison with Table 1 and Table 2. A graphical representation of the comparison of the average identification rate of Four-scripts, Tri-scripts and Bi-scripts combinations are shown in Fig. 11.

Table 3. Bi-script identification rate using MLP classifier on the test dataset of ${}^4C_2$ sets.

Sl. no.	Scripts combination	AAR (%)
1	{Bangla, Urdu}	90.9
2	{Devanagari, Urdu}	89.6
3	{Roman, Urdu}	82.1
4	{Bangla, Roman}	80.2
5	{Devanagari, Roman}	78.6
6	{Bangla, Devanagari}	71.8
7	**Avg. Bi-script Acc. Rate**	**82.2%**

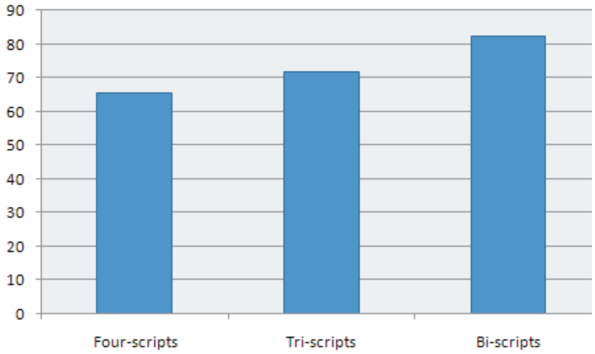

Fig. 11. Average accuracy rate of different script combination patterns i.e. Four-scripts, Tri-scripts and Bi-scripts using MLP classifier.

3.5 *Comparative study*

Exact comparative study with the work of other fellow researches is not possible right now as no work is reported on *HNSI* considering Bangla, Devanagari, Roman and Urdu scripts. Availability of benchmark database is another problem in this field, that's why we have taken the effort to prepare our own image dataset. The present result can be considered as a benchmark for these Bi-scripts, Tri-scripts and Four-script combinations.

4. Conclusion and Future Scope

Handwritten *NSI* problem was never been addressed before for a multi-script country like India. In this paper a new framework for *NSI* from handwritten document images has been proposed based on combination of daubechis wavelet decomposition and spatial domain features. Four popular eastern Indian scripts namely Bangla, Devanagari, Roman and Urdu numeral images are considered for experimentation. Due to unavailability of standard numeral dataset, a new numeral string image dataset of these four scripts has been prepared. An easy to calculate and robust multi-dimensional feature set is constructed for training the classifier. Performance of different well known classifiers has been evaluated and benchmark result is developed on the test dataset for Four-scripts, Tri-scripts and Bi-scripts combinations using MLP classifier. The under performing script combinations are identified and this issue will be taken care in future work. Scopes can be further extended to develop numeral image dataset for all official Indic scripts and their bench marking with respect to *HNSI* problem. Finally it is worth to be mentioned here, that the present dataset (4000 handwritten numeral word images of Bangla, Devanagari, Roman and Urdu) will be provided freely on request for only non-commercial use.

Acknowledgements

The authors are very much thankful to Mr. Tousif Jaman and Mr. Sahaniaj Dhukra, B.Sc. final year students of Aliah University for their sincere and unconditional help during data collection process.

References

[1] S. M. Obaidullah, S. K. Das and K. Roy, A System for Handwritten Script Identification from Indian Document, In *Journal of Pattern Recognition Research*, vol. 8 no. 1, pp. 1–12, (2013).

[2] D. Ghosh, T. Dube and S. P. Shivprasad, Script Recognition – A Review, *IEEE Trans. on Pattern Analysis & Machine Intelligence*, vol. 32, no. 12, pp. 2142–2161, December (2010).

[3] B. B. Chaudhuri and U. Pal, A Complete Printed Bangla OCR, Pattern Recognition, vol. 31, pp. 531–549, (1998).

[4] S. Vajda, K. Roy, U. Pal, B. B. Chaudhuri and A. Belaid, Automation of Indian Postal Documents Written in Bangla and English, *Int. Journal of Pattern Recognition and Artificial Intelligence*, vol. 23, no. 8, pp. 1599–1632, (2009).

[5] J. Hochberg, P. Kelly, T. Thomas and L. Kerns, Automatic Script Identification from Document Images using Cluster-based Templates, In *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 176–181, (1997).

[6] Santanu Chaudhury, Gaurav Harit, Shekar Madnani and R. B. Shet, Identification of Scripts of Indian Languages by Combining Trainable Classifiers, In *Proceedings of Indian Conference on Computer Vision, Graphics and Image Processing*, December 20–22, Bangalore, India, (2000).

[7] D. Dhanya, A. G. Ramakrishnan and Peeta Basa Pati, Script Identification in Printed Bilingual Documents, In *Sadhana*, vol. 27, part-1, pp. 73–82, (2002).

[8] P. B. Pati and A. G. Ramakrishnan, Word Level Multi-Script Identification, *Pattern Recognition Letters*, vol. 29, no. 9, pp. 1218–1229, (2008).

[9] S. M. Obaidullah, A. Mondal, N. Das and K. Roy, Script Identification from Printed Indian Document Images and Performance Evaluation Using Different Classifiers, *Applied Computational Intelligence and Soft Computing*, Article ID 896128, vol. 2014, p. 12, (2014). doi:10.1155/2014/896128

[10] J. Hochberg, K. Bowers, M. Cannon and P. Kelly, Script and Language Identification for Handwritten Document Images, In *International Journal of Document Analysis and Recognition*, vol. 2, issue 2–3, pp. 45–52, (1999).

[11] L. Zhou, Y. Lu and C. L. Tan, Bangla/English Script Identification Based on Analysis of Connected Component Profiles, *Lecture Notes in Computer Science*, vol. 3872/2006, pp. 243–254, (2006).

[12] V. Singhal, N. Navin and D. Ghosh, Script-based Classification of Hand-written Text Document in a Multilingual Environment, *Research Issues in Data Engineering*, pp. 47–54, (2003).

[13] M. Hangarge, K. C. Santosh and R. Pardeshi, Directional Discrete Cosine Transform for Handwritten Script Identification, In *Proceedings of* 12th *International Conference on Document Analysis and Recognition*, pp. 344–348, (2013).

[14] R. Pardeshi, B. B. Chaudhury, M. Hangarge and K. C. Santosh, Automatic Handwritten Indian Scripts Identification, In *Proceedings of* 14th *International Conference on Frontiers in Handwriting Recognition*, pp. 375–380, (2014).

[15] U. Pal, R. K. Roy, K. Roy and F. Kimura, Indian Multi-Script Full PIN Code Recognition for Indian Postal Automation, In *10th International Conference on Document Analysis and Recognition, ICDAR 2009*, Barcelona, Spain, (2009).

[16] K. Roy, A. Banerjee and U. Pal, A System for Word-wise Handwritten Script Identification for Indian Postal Automation, In *Proceedings of IEEE India Annual Conference, 2004*, pp. 266–271, (2004).

[17] K. B. Raja, S. Sindhu, T. D. Mahalakshmi, S. Akshatha, B. K. Nithin, M. Sarvajith, K. R. Venugopal and L. M. Patnaik, Robust Image Adaptive Steganography using Integer Wavelets, In *Proceedings of International Conference on information Processing (ICIP 2007)*, pp. 100–106, Bangalore, August 10–12, (2007).

[18] B. B. Chaudhuri and U. Pal, A Complete Handwritten Numeral Database of Bangla – A Major Indic Script, 10th *International Workshop on Frontiers in Handwriting Recognition*, (2006).

[19] K. G. Srinivasa, K. Sridharan, P. D. Shenoy, K. R. Venugopal and L. M. Patnaik, An Efficient Neural Network Based CBIR System Using STI Features and Relevance Feedback, *International Journal on Intelligent Data Analysis, IOS Press*, vol. 10, no. 2, pp. 121–137, June (2006).

[20] http://in.mathworks.com/products/datasheets/pdf/wavelet-toolbox.pdf, accessed on 01.01.2015