Middle-East Journal of Scientific Research 22 (10): 1514-1519, 2014
ISSN 1990-9233

DOI: 10.5829/idosi.mejsr.2014.22.10.21487

Feature Extraction and Dimensionality Reduction in Pattern Recognition Using Handwritten Odia Numerals

[1]*Pradeepta K. Sarangi* and [2]*Kiran K. Ravulakollu*

[1]Apeejay Institute of Technology, Greater Noida, U.P, India
[2]Sharda University, Greater Noida, U.P, India

Abstract: Feature extraction is the initial and critical stage that needs to be carried out for any recognition system that uses pattern matching. Many of the existing feature extraction techniques such as Quadrant-mean, Histogram and Texture extraction are based on the principle of partitioning the image into small segments. However, as the size of the image increases, the complexity of computation also increases. When it comes to image reconstruction from the feature values, these methods are not successful in generation a complete and valid original image. In addition, as the number of components in the image increases, the dimensionality of the features also increases. In order to reduce the feature extraction complexity, dimensionality reduction is applied to increase the recognition performance. To perform image restoration, transformation of limited feature vector to binary image is considered. In this paper, dimensionality reduction and feature extraction is achieved using row-wise decimal conversion for pattern recognition. This signifies the transformation of features (small number) to a normalized state thereby preserving quality of image (numeral). A simple recurrent neural network (RNN) has been used as the classifier and recognition accuracy is reported. The method has been implemented on the dataset consisting of 1500-isolated handwritten Odia numerals demonstrating an accuracy of 92.4%. Experimental results show that the proposed method has the potential to be used as a feature extraction technique for handwritten Odia numerals.

Key words: Handwritten Recognition · Odia Numerals · Recurrent Neural Network · Feature Extraction and Pattern Recognition

INTRODUCTION

Odia is a regional language derived from the Devanagari script and commonly used in north-eastern States of India. It is one of the many official languages of India and mainly spoken in Odisha and in some parts of West Bengal. Though Odia is one of many official languages, still research on Odia character recognition system is not so advanced like on other Indian languages. The condition for handwritten script is even worse. Whatever research works have been done so far or the ongoing researches are moreover about the implementation of various existing techniques either single or combined. Research on handwritten Odia characters has not been much explored as only a few research centers at national level are involved in exploring Odia script. Utkal University, Bhubaneswar is the only center in the State (Odisha) reported to be engaged in research on Odia script [1, 2]. Hence, a lot of scope is available for researchers in this area. Here, we present some of the related and recent developments on handwritten Odia numerals by various researchers (table-1) including our own previous works [3, 4].

Dataset is another key factor for any successful OCR system design. Availability of a standard dataset helps researchers to easily implement new methods /techniques in the field and compare results. In case of Odia numerals, no knowledgeable standard dataset is available in public domain. Researchers have to develop their own dataset as required. However, ISI Kolkata has developed a dataset on

Corresponding Author: Pradeepta K. Sarangi, Apeejay Institute of Technology, Greater Noida, U.P, India.

Table 1: Summary of related and recent works

Authors	Classifier	Recognition Accuracy (%)
Sarangi *et al*. [3]	Naive Bayes	92.75
Sarangi *et al*.[4]	Neural Network	85.30
Jindal *et al*.[5]	MLP	94.20
Mishra *et al*. [6]	BPNN	92.00
Sarangi *et al*.[7]	Hop Field	95.40
Mahato *et al*. [8]	ANN	93.20
Pal *et al*. [6]	MQC	98.40
Bhowmik *et al*.[10]	HMM	90.50
Roy *et al*. [6]	Quadratic	90.38

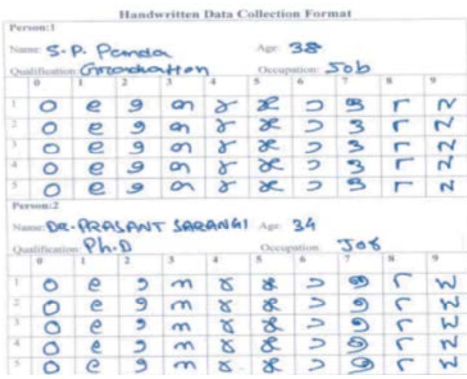

Fig 1: Sample dataset collected from two persons

handwritten Odia numerals, which is available on request. In this research, we have used our own collection of data. We have collected handwritten numerals from several people on a plain paper. The respondents were of different qualifications, age groups and professions. Each respondent was asked to write each numeral five times as shown in the figure 1.

A total of 1500 different numerals for ten classes (0 to 9) consisting of 150 numerals of each class have been collected from 30 people. The dataset has been divided into training patterns and test patterns.

Feature Extraction: Feature extraction is the process of finding a smaller set of elements that represent the original object so that the computerized experiments could be achieved faster and with fewer memories requirement. Feature vector represents the inherent properties of the original object. Too many parameters in the feature vector may not solve the purpose at all and at the same time very few parameters may not be able to represent the original object in appropriate form. Feature extraction is highly subjective in nature and depends on the type of problem we are trying to handle. No generic feature extraction method is available to work for all cases. It is also almost impossible to rank an algorithm as the best for feature selection or extraction. It all depends on the application at hand. Since no standard feature extraction

Table 2: Feature extraction techniques in Odia OCR

Authors	Features
Sarangi *et al*. [3,4]	LU Factors
Jindal *et al*.[5]	Zernike moment
Mishra *et al*. [6]	DCT and DWT
Sarangi *et al*. [7]	Binary image
Mahato *et al*. [8]	Quadrant mean
Pal *et al*. [9, 10]	Directional
Bhowmik *et al*. [10]	Scalar
Roy *et al*. [11]	Chain Code Histogram

method exists, researchers have to develop or adopt suitable feature extraction method depending on the nature and properties of the dataset. A list of various feature extraction methods used in various pattern recognition systems is found in [2]. In case of handwritten Odia numeral recognition system, no knowledgeable work is reported towards proposition of new feature extraction method. The authors have mainly put their efforts towards using existing feature extraction techniques except our own work [3, 4]. Here, we present some of the feature extraction methods used in handwritten Odia numeral recognition [Table-2].

During the analysis of the existing feature extraction methods used in handwritten Odia numerals, we came across some basic questions in our mind like:

- Are all feature extraction methods based on mathematical/logical derivations?
- Is minimum size of the image a relevant factor for particular feature extraction method?
- What relationship exists between the feature vectors and the original image?
- Is it possible to reconstruct the original image from the feature vectors?

Non-availability of sufficient literatures to give satisfactory answer to these questions motivated us to propose a new feature extraction method, which could probably answer these questions maintaining the basic objectives of feature extraction along with efficiency and accuracy.

Proposed Method of Feature Extraction: The proposed method is based on the row wise decimal conversion of the elements using binary matrix of the image. The block diagram of the proposed method is given in figure-2

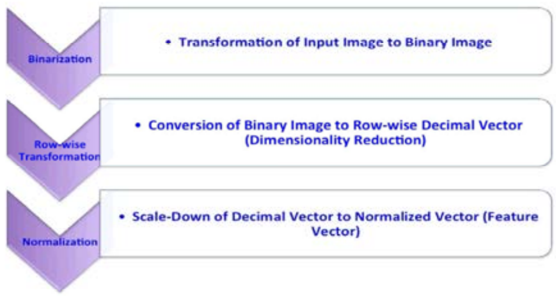

Fig 2: Block diagram of the proposed method

The main characteristics of this method are:

- It is very simple to calculate the feature values. Only three steps are required to extract the feature values.
- The feature values are extracted by a set of inter-connected logical steps.
- The size of the feature vector depends directly on the size of the image. If the image-size increases, then the size of feature vector also increases. However, there is no need of using large image size as an image of size as low as of '8 x 8' could effectively represent the original image.
- The original image could be reconstructed from the feature vector by back-tracking the steps.
- A one-to-one relationship exists between the original image and the feature vector.

The algorithm for this proposed method is given as below:

Algorithm for this proposed method
1. Pre-process the extracted numeral image (cropping & resizing).
2. Convert the input gray scale image to binary image.
3. Calculate the row wise decimal values considering one row in the binary image as one binary number.
4. Finally, scale down the decimal values to the range of 'zero' to 'one' using suitable formula.

Here, we have used the following formula:

$$\text{Scaled Down Value}=(\text{Actual Value-X})*\frac{Y}{Z} \quad (1)$$

where, the X is the lower bound of the data set, Y is the value calculated based on the difference of upper bound and lower bound of data set and Z is one unit based on the value of Y. Each image has a fixed binary representation. If we take each row in the binary image as a binary number and convert it to decimal values then we will get a set of decimal values. Each of these decimal values represents one row in the binary image. For example, if we take an image size of '8 X 8' then we will

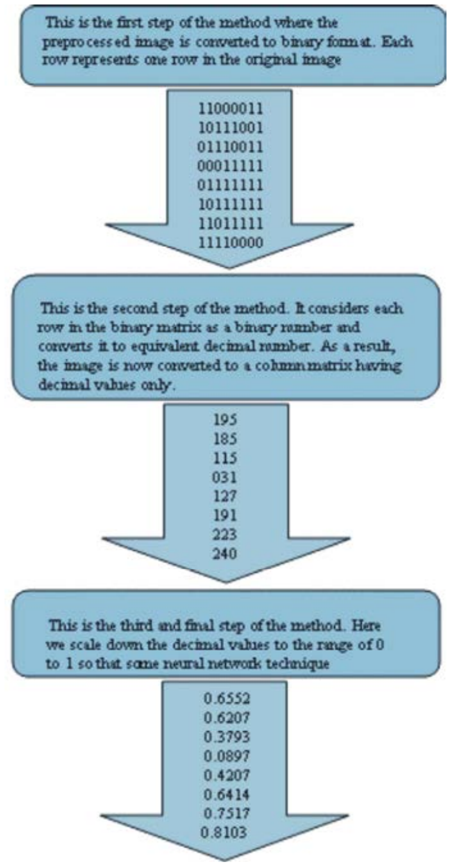

Fig 3: Feature extraction using the proposed method

get 'eight' rows of decimal values. Then converting these decimal values to the range 'zero' to 'one', we will get the desired feature vector. An explanation is given in figure 3.

The above diagram describes a pictorial representation of an example of implementation of the proposed feature extraction method.

Implementation Design & Results' Analysis: Selection of a suitable classifier is an important aspect in the performance of any OCR system. However, no standard rules are available to decide the classifier. All these have to be done only on experimental basis based on the nature of the script and characters. Since this research deals with handwritten Odia numerals, so the implementation strategy has been designed keeping in mind the nature of the dataset. A simple recurrent neural network (also known as Elman Network) has been used as the classifier for this research. An Elman network is an MLP with a single hidden layer and in addition it contains connections from the hidden

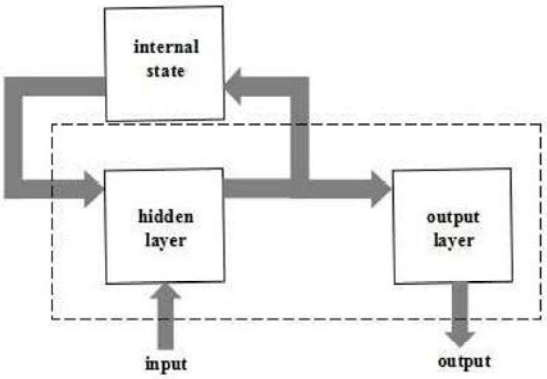

Fig 4: A simple Elman Network

layer's neurons to the context units. The context-units store the output values from the hidden neurons in a time unit and these values are fed as additional inputs to the hidden neurons in the next time unit. RNNs use their internal memory to process arbitrary sequences of inputs. This makes them applicable to handwriting recognition, where they have achieved the best-known results [12]. The main characteristic of RNN is the internal feedback or time-delayed connections. A simple structure of RNN [13] is given in figure 4.

The diagram above represents a simple structure of an Elman network. The dotted box (lower one) represents a simple MLP and the upper box (out side the dotted line) represents the internal states that store the time delay.

Training Algorithm: Training procedure of a recurrent neural network is similar to the case of MLP training where the network's output is compared with the target output. The square error is used to update the network's weights according to the error back propagation algorithm.

If X^n is the vector produced by the union of input and context vectors, then the training algorithm for an Elman network is very similar to the algorithm for MLP network training and is described as below [13]:

- Initialize the weight vector *w(0)* with random values in (1, 1), the learning rate η, the repetitions counter (*k=0*). Initialize the context nodes at 0.5.
- Let *W(k)* be the network's weight vector in the beginning of epoch *k*

- Start of epoch *k*. Store the current values of the weight vector $W_{old}=W(k)$
- For *n=1, 2, 3, …N*
- Select the training example (X^n,t^n) and apply the error back propagation in order to compute the partial derivatives $\frac{\partial E^n}{\partial w_i}$
- Update the weights $w_i(k+1) = w_i(k) - \eta \frac{\partial E^n}{\partial w_i}$
- Copy the hidden nodes' values to the context units.

- *End of epoch* k Termination check. If true, terminate.
- *k=k+1.* Go to step-2.

Implementation: A simple recurrent neural network (SRN) with back propagation learning has been implemented in Matlab-7 on a standard Windows operating system environment. The network structure and various parameters used in this implementation are summarized below.

Neural Network Specification
Architecture: 8-8-10
Type: Recurrent neural network
Input patterns: Feature vectors (8 elements)
No. of test patterns: 1200
Learning algorithm: Back propagation
Performance: Mean Squared Error (MSE).

An example of the network structure is given in figure-5.

The diagram above represents a simple structure of a MLP consisting of one input layer (8 elements), one hidden layer (8 elements) and one output layer (10 elements).

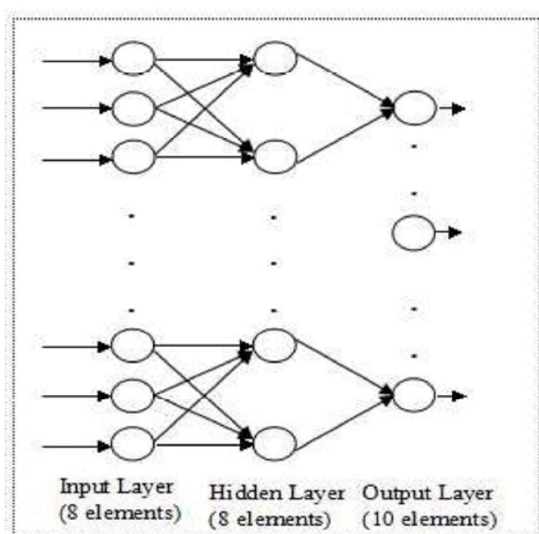

Fig 5: An example of the network structure

Table 3: Network output for test patterns

Input Class	Correct Classifications	Incorrect Classifications	Recognition Accuracy
0	113	7	94.16
1	102	18	85.00
2	105	15	87.50
3	108	12	90.00
4	110	10	91.66
5	116	4	96.66
6	116	4	96.66
7	109	11	90.83
8	116	4	96.66
9	114	6	95.00

Table 4: Confusion Matrix

Class	0	1	2	3	4	5	6	7	8	9
0	113	4	3	0	0	0	2	0	0	0
1	11	102	12	0	2	0	3	0	0	0
2	2	11	105	0	0	0	0	2	0	0
3	0	1	2	108	4	0	2	0	1	2
4	1	2	1	0	110	3	0	2	1	0
5	04	0	0	0	3	116	0	1	0	1
6	1	0	2	0	0	0	116	0	0	1
7	1	1	43	2	0	0	2	109	0	1
8	0	0	0	0	0	0	1	0	116	3
9	0	0	0	0	0	0	1	1	4	114

RESULTS AND DISCUSSION

Once the training was over, the target patterns were presented to the network one by one and the output was noted. The results are summarized in Table 3.

A total of 1200 patterns for all ten classes (120 patterns for each class) have been tested on the network. The confusion matrix is given in table 4 below.

Here, we observe that the incorrect classifications are more between '1' and '2' and reason is that the characters 1 and 2 in Odia language are just mirror image to each other. This problem could be resolved if we consider the column decimal values instead of row decimal values.

CONCLUSION

An experiment has been done with a new feature extraction method for handwritten Odia numerals. An overall recognition accuracy of 92.41% is reported when the proposed method has been implemented using recurrent neural network as the classifier. When we compare the recognition accuracy with other similar works (Table 1), this result stands at fifth best position. This does not imply that this method is inefficient. The recognition accuracy differs for different languages even when same feature extraction and classification techniques are used. Here we just propose a new method and it is in the beginning stage. It is also too early to say that the method will be universally applicable for all types of scripts. More experiments and researches are required to test the suitableness of this method applicable for other languages. Even if for handwritten Odia numerals, the applicability of this proposed feature extraction method needs to be tested with a larger size data set and with Odia alphabets also.

REFERENCES

1. Pal, U., R. Jayadevan and N. Sharma, 2012. Handwriting Recognition in Indian Regional Scripts: A Survey of Offline Techniques. ACM Transactions on Asian Language Information Processing, 11(1): 1-36.
2. Sahoo, A.K., G.S. Mishra and K.K. Ravulakollu, 2014. Sign Language Recognition: State of the Art. ARPN Journal of Engineering and Applied Sciences, 9(02): 116-134.
3. Sarangi, P.K., P. Ahmed and K.K. Ravulakollu, 2014. Naïve Bayes Classifier with LU Factorization for Recognition of Handwritten Odia Numerals. Indian Journal of Science and Technology, 7(1): 35-38.
4. Sarangi, P.K. and P. Ahmed, 2013. Recognition of Handwritten Odia Numerals Using Artificial Intelligence Techniques. The International Journal of Computer Science & Applications, 2(02): 41.
5. Jindal, T. and U. Bhattacharya, 2013. Recognition of Offline Handwritten Numerals Using an Ensemble of MLPs Combined by Adaboost. MOCR-13,ACM, Washington, DC, USA. Article, 18(978-1-4): 503-2114-3.
6. Mishra, T.K., B. Majhi and S. Panda, 2013. A Comparative Analysis of Image Transformations for Handwritten Odia Numeral Recognition. Advances in Computing, Communications and Informatics (ICACCI), 2013 International Conference on.. ISBN: 978-1-4799-2432-5, pp: 790-793.
7. Sarangi, P.K., A.K. Sahoo and P. Ahmed, 2012. Recognition of Isolated handwritten Numerals Using Hopfield Neural Network. International Journal of Computer Applications, 40(8): 36-42.

8. Mahato, M.K., A. Kumari and S.C. Panigrahi, 2011. A System for Oriya Handwritten Numeral Recognition for Indian Postal Automation. IJASTRE, 1: 1.
9. Pal, U., T. Wakabayashi, N. Sharma and F. Kimura, 2007. Handwritten Numeral Recognition of Six Popular Indian Scripts. In proceedings of 9th ICDAR, pp: 749-753.
10. Bhowmik, T.K., S.K. Parui, U. Bhattacharya and B. Shaw, 2006. An HMM Based Recognition Scheme for Handwritten Oriya Numerals. In Proceedings of 9th ICIT, pp: 105-110.
11. Roy, K., T. Pal, U. Pal and F. Kimura, 2005. Oriya Handwritten Numeral Recognition System. in Proceedings of ICDAR, pp: 770-774.
12. Graves, A., M. Liwicki, S. Fernandez, R. Bertolami, H. Bunke and J. Schmidhuber, 2009. A Novel Connectionist System for Improved Unconstrained Handwriting Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 31: 5.
13. Salatas, J., 2011. Implementation of Elman Recurrent Neural Network in WEKA. Downloaded from http://jsalatas.ictpro.gr/implementation-of-elman-recurrent-neural-network-in-weka/.