Neural Network based Kannada Numerals Recognition System

Shreedharamurthy S K
Asst, Professor, E&C Dept.
University BDT College of Engineering
Davangere-577004, Karnataka-India

H.R.Sudarshana Reddy
Phd, Professor, & Head E&E Dept.
University BDT College of Engineering
Davangere-577004, Karnataka-India

ABSTRACT

This paper presents a novel approach for feature extraction in spatial domain to recognize segmented (isolated) Kannada numerals using artificial neural networks. Artificial neural systems represent the promising new generation of information processing networks to develop intelligent machines which can be used as classifier. The ability of neural networks to learn by ordinary experience, as we do, and to take sensitive decisions give them the power to solve problems found intractable or difficult for traditional computation. In this paper, the development of handwritten Kannada numeral recognition system using spatial features and neural networks is reported. Handwritten numerals are scan converted to binary images and normalized to a size of 30 x 30 pixels. The features are extracted using spatial co ordinates and are classified successfully using the feed forward neural network classifier.

Keywords:

Handwritten Kannada numerals, Artificial Neural Network, , Feature Extraction, Pattern Classification.

1. INTRODUCTION

Pattern recognition encompasses two fundamental tasks: description and classification. Given an object to analyze, a pattern recognition system first generates a description of it (i.e., the pattern) and then classifies the object based on that description (i.e., the recognition).

Character extraction and recognition techniques have potential application in any domain where a large mass of document image-bearing texts must be interpreted or analyzed. Conventionally, such images are processed by human operators who act according to what has been written or simply key in what they read onto a computer system that carries out further processing, say of postal address [1] This process can be automated using computer popularly known as pattern recognition system.

The essential problem of pattern recognition is to identify an object as belonging to a particular group. Assuming that the objects associated with a particular group share common attributes more so than with objects in other groups, the problem of assigning an unlabeled object to a group can be accomplished by determining the attributes of the object (i.e., the pattern) and identifying the group of which those attributes are most representative (i.e., the recognition). This can be done by artificial neural networks.

The main thing missing in computers when compared to humans is the ability to think and take decisions as human brain does. One of the ways of making computers brainy is to simulate a human brain in the computer. The brain learns by experience (i.e., through examples). In order to simulate a brain in the computer, we have to make the computer to learn by examples and use this knowledge in future.

Pattern recognition is becoming more and more important in modern world. It helps humans ease their jobs and solve more complex problems. The recognition of handwritten numerals has been the subject of much attention in pattern recognition because of its number of applications such as bank check processing, interpretation of ID numbers, vehicle registration numbers and pin codes for mail sorting.

The penetration of Information Technology (IT) becomes harder in a country such as India where the majority people read and write in their native language. Therefore, enabling interaction with computers in the native language and in a natural way such as handwriting is absolutely necessary [4]. The need for OCR arises in the context of digitizing Kannada documents from the ancient and old era to the latest, which helps in sharing the data through the Internet [5]. Kannada, the native language of a southern state in India i.e, Karnataka has several million speakers across the world and is received cultural status from central government very recently.

The Kannada language is one of the four major south Indian languages. It is spoken by about 50 million people in the Indian states of Karnataka, Tamilnadu, Andhra Pradesh and Maharashtra. The Kannada alphabet consists of 16 vowels and 36 consonants. It also includes 10 different symbols representing the ten numerals of the decimal number system as shown in figure1.

Figure1. Kannada numerals 0 to 9

In the literature, many papers have been published with research detailing new techniques for the classification of handwritten characters and promising feature extraction methods have been identified in the literature for recognition of characters and numerals of many different scripts. These include template matching, projection histograms, geometric moments, Zernike moments, contour profile, Fourier descriptors, and unitary transforms. A brief review of these feature extraction methods is found in [1]. Various methods have been proposed, and high recognition rates are reported, for the recognition of English handwritten digits [8 - 10]. The task of classification is to partition the feature space into regions corresponding to source classes or assign class confidences to each location in the feature space.

Statistical techniques, neural networks, and more recently support vector machine (SVM) have been widely used for classification due to the implementation efficiency [1-5].

From the literature survey, it is evident that the work on handwritten Kannada numeral recognition is still in infant stage. This has motivated us to design a recognition system for handwritten Kannada numerals.

2. NUMERALS RECOGNITIN SYSTEM

A typical pattern recognition system consists of three stage processes as shown in figure 2. The first stage is Pre-processing, second stage is Feature extraction and the third.

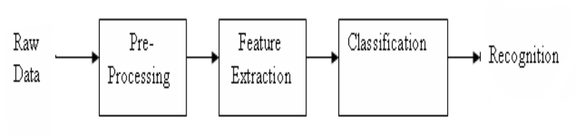

Figure 2. Pattern recognition system

2.1 Pre-processing

Pre-processing involves normalizing the raw data given to the computer so that the further processing is easier. The typical preprocessing operations involves noise reduction, size normalization, slant estimation and correction, thinning, segmentation etc.,

2.2 Feature Extraction

When the input data to an algorithm is too large to be processed and it is suspected to be redundant then the input data will be transformed into a reduced representation set of features. Transforming the input data into the set of features is called *features extraction*. If the features extracted are carefully chosen it is expected that the features set will extract the relevant information from the input data in order to perform the desired task using this reduced representation instead of the full size input. The set of features that are used makes up a feature vector, which represents each member of the population. Then, pattern recognition system classifies each member of the population on the basis of information contained in the feature vector [6].

Selection of a feature extraction method is probably the single most important factor in achieving high recognition performance in pattern recognition systems. The features that are extracted are fed to the classifiers for further processing. Spatial centroid feature vectors are used to classify the Kannada numerals in our system.

2.2.1 Spatial Centroid Features

Feature extraction involves simplifying the amount of resources required to describe a large set of data accurately. When performing analysis of complex data, one of the major problems stems from the number of variables involved. Analysis with a large number of variables generally requires a large amount of memory and computation power or a classification algorithm which over fits the training sample and generalizes poorly to new samples.

Feature extraction is a general term for methods of constructing combinations of the variables to get around these problems while still describing the data with sufficient accuracy. Best results are achieved when an expert constructs a set of application-dependent features.

Proposed method:

In the proposed method the captured image (hand written numeral) has to be binarized so that the numeral images have pixel values 0 and 1. Each numeral image represents a numeral (binary 1) that is unconstrained, isolated and clearly discriminated from the background (binary 0). This can be done by converting the captured image to bmp format and then preprocessing of the same is done.

The bmp format of image of Kannada numeral 1 contains only zeroes and ones and is as shown in figure 3.

0 0 0 0 **1 1 1 1** 0 0 0 0

0 0 0 **1** 0 0 0 0 **1** 0 0 0

0 0 **1** 0 0 0 0 0 0 **1** 0 0

0 0 **1** 0 0 0 0 0 0 **1** 00

0 0 **1** 0 0 0 0 0 0 0 **1** 0

0 0 **1** 0 0 0 0 0 0 0 **1** 0

0 0 **1** 0 0 0 0 0 0 0 **1** 0

Figure 3. BMP Image values

The features from the bmp image can be obtained by reading the spatial coordinate (x,y) values of the pixels having the value '1'only. The centroid of the two coordinate values x and y are obtained by taking the average of those coordinates i.e,

Centroid = (x+y)/2

Number of centroids obtained in this way depends on the number of one's present in the corresponding bmp image. Variable length centroids are generated for different numerals and are expected to be same numbers for all the classes. This can be done by equisampling the all numeral bmp images or otherwise padding the zeroes for the difference at the end. These centroids are treated as the feature vectors in our system.

The algorithm for generating the spatial centroid features is given below:

Algorithm

1. Read the pattern in bmp format.

2. Start from top left scan image line by line.

3. Note down the spatial coordinate values (x,y) of first 'ON'cell in the first line

4. Feature value F (centroid) is the average of coordinates x and y i.e, F = (x+y)/2

5. Obtain remaining features repeating above step for all 'ON' cells in the image.

2.3 Classification

Classification is a step in numerals recognition which accepts extracted features from the feature extraction step and identifies the pattern written. A large number of classifiers are available: parametric and nonparametric statistical classifiers, neural networks, support vector machines (SVMs), hybrid classifiers etc [9]. Artificial Neural Network has been used as a classifier in our system.

2.3.1 Artificial neural network (ANN)

Artificial neural network systems have great ability to learn by experience and generalize the inputs to produce reasonable outputs for inputs that were not encountered during learning (training).

2.3.1.1 The Multi-Layer Perceptron

Multi-layer perceptrons are one of many different types of existing neural networks. They comprise a number of neurons connected together to form a network. The strength or a weight of the links between the neurons is where the functionality of the network resides. Its basic structure is shown in figure 4.

The idea behind neural networks stems from studies of the structure and function of the human brain. Neural networks are useful to model the behaviors of real-world phenomena. Being able to model the behaviors of certain phenomena, a neural network is able subsequently to classify the different aspects of those behaviors, recognize what is going on at the moment, diagnose whether this is correct or faulty, predict what it will do next, and if necessary respond to what it will do next.

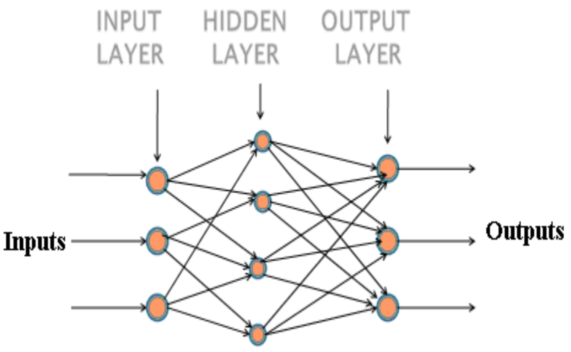

Figure 4. Feed forward Neural Network

2.3.1.2 Feed Forward Back Propagation Network

Feed forward networks often have one or more hidden layers of sigmoid neurons followed by an output layer of linear neurons. Multiple layers of neurons with nonlinear transfer functions allow the network to learn nonlinear and linear relationships between input and output vectors. Back-propagation learning rule is used train to multiple-layer networks. Input vectors and the corresponding target vectors are used to train a network until it can approximate a function, associate input vectors with specific output vectors. Networks with biases, a sigmoid layer, and a linear output layer are capable of approximating any function with a finite number of discontinuities.

2.3.1.3 Forward Propagation

Forward propagation is the process whereby each of all of the neurons calculates its output value, based on inputs provided by the output values of the neurons that feed it. The input neuron distributes the signal along multiple paths to hidden layer neurons.

A weight is associated to a hidden neuron. Each node of input layer is connected to every node of hidden layer. Likewise each node of hidden layer is connected to every node of output layer again by some weights as shown in figure 5. Also the data flows from left to right. Hence the network is called feed forward network. The output of a neuron is a function of its net input. This function can be trigonometric, hyperbolic or sigmoid function.

2.3.1.4 Error Back Propagation Learning

Back propagation learning algorithm is popularly used to train a feed forward neural network. The error back propagation consists of two passes through the different layers of the network; a forward pass and a backward pass.

Forward pass is same as the forward propagation. Back propagation is an iterative process that starts with the last layer and moves backwards through the layers until the first layer is reached.

For each set of inputs, a set of target values is provided. During learning, the difference between the set of output and target values is found to get the error value. After the feed forward process, this error is back propagated and weights between the layers are adjusted starting from output layer to hidden layer and then from hidden layers to input layer to minimize the error. This process is repeated till it reaches the required minimum error value.

2.4 Training

The process of training is preparing the Artificial Neural Network to recognize the desired set of characters. For character to be recognized, a set of similar characters with different size and little variation in their shape is written and used for training. Standard back-propagation is a gradient descent algorithm, in which the network weights are moved along the negative of the gradient of the performance function. The term back-propagation refers to the manner in which the gradient is computed for nonlinear multilayer networks. There are a number of variations on the basic algorithms those are based on other standard optimization techniques, such as conjugate gradient and Newton methods.

With standard steepest descent, the learning rate is held constant throughout training. The performance of the algorithm is very sensitive to the proper setting of the learning rate. If the learning rate is set too high, the algorithm may oscillate and become unstable. If the learning rate is too small, the algorithm will take too long to converge. It is not practical to determine the optimal setting for the learning rate before training, and, in fact, the optimal learning rate changes during the training process, as the algorithm moves across the performance surface.

The gradient descent algorithm for training the multi-layer perceptron was found slow especially when getting close to a minimum (since the gradient is disappearing). One of the reasons is that it uses a fixed-size step. In order to take into account the changing curvature of the error surface, many optimization algorithms use steps that vary with each iteration. In order to solve this problem, an adaptive learning rate can be applied to attempt keeping the learning step size as large as possible while keeping learning stable. The learning rate is made responsive to the complexity of the local error surface. In this approach, new weights and biases are calculated using the current learning rate at each epoch. New outputs and errors are then calculated. As with momentum, if the new error exceeds the old error by more than a predefined ratio for example, 1.04, the new weights and biases are discarded. In addition, the learning rate is decreased.

Otherwise, the new weights are kept. If the new error is less than the old error, the learning rate is increased. This procedure increases the learning rate.

In our system, a feed-forward multi-layer perceptron with a single hidden layer and trained by gradient descent with momentum and a learning rate back-propagation method was applied to the digit classification problem

2.5 Recognition

Once the artificial neural network is trained to recognize a set of numerals, it is ready to use for recognizing digits in the numerals recognition system. During recognition phase, Artificial Neural Network has the capacity to generalize and identify the numerals written with little variations when compared to the numerals used for training

3. RESULTS AND CONCLUSION

The proposed system describes a novel procedure which uses spatial features and artificial neural network as classifier. to recognize hand written Kannada numerals. We have used 100 samples of numerals from the created data base, sample patterns of which shown in figure 5. Out of which 80 patterns used for training phase and 20 samples for testing phase. We achieved around 95% of recognition rate as illustrated in the table 1.

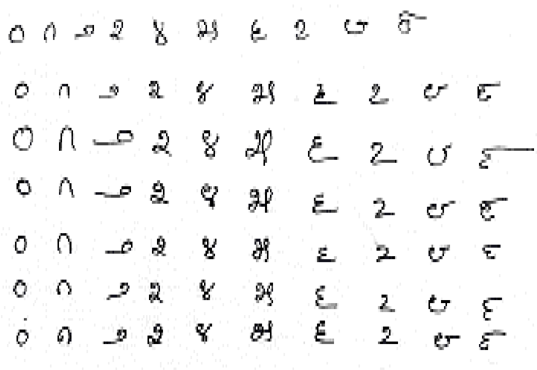

Figure 5. A sample patterns of Kannada Handwritten numerals 0 to 9

Table.1: Recognition Rate

Numerals	Percentage of recognition
0	100
1	95
2	100
3	95
4	85
5	95
6	95
7	100
8	85
9	95
Overall percentage:	94.5

4. REERENCES

[1] K V Prema_ and N V Subbareddy- Two-tier architecture for unconstrained handwritten character recognition- *Sadhana* Vol. 27, Part 5, October 2002, pp.585-594.

[2] Hyun-Chul Kim, Daijin Kim, Sung Yang Bang-A numeral character recognition using PCA mixture model, pattern recognition letters, Vol 23, 2002, pp.103-111

[3] G Y Chen, T D Bui, A. Krzyzak-Contour based numeral recognition using mutiwavelets and neural networks, pattern recognition letters, Vol 36, 2003, pp.1597-1604

[4] Rejean Plamondon, Sargur.N.Srihari, On-line and Off-line Handwriting Recognition: A Comprehensive survey, IEEE *Trans,Pattern Analysis and Machine Intelligence, vol 22, no 1, pp 63*-79,Jan 2000

[5] Claus Bahlmann-Directional features in online handwriting recognition, pattern recognition letters, Vol 39, 2006, pp.115-125

[6] Oivind Due Trier, Anil.K.Jain and Torfinn Taxt,-Feature Extraction Methods for Character Recognition – A Surve, July 1995.

[7] Alexander Goltsev, Dmitri Rachkovskij-Combination of the assembly neural network with a perceptron for recognition of handwritten digits arranged in numeral strings, pattern recognition letters, Vol 38, 2005, pp.315-322

[8] Bailing Zhang, Minyue Fu, Hong Yan-A nonlinear neural network model of mixture of local principal component analysis: application to handwritten digits recognition, pattern recognition letters, Vol 34, 2001, pp.203-214

[9] Cheng-Lin Liu, Kazuki Nakashima, Hiroshi Sako, Hiromichi Fujisawa, Handwritten digit recognition: investigation of normalization and feature extraction techniques, Jun 2003

[10] R. Jagadeesh Kannan and R. Prabhakar- Off-Line Cursive Handwritten Tamil Character Recognition- WSEAS TRANSACTIONS on SIGNAL PROCESSING- ISSN: 1790-5052 Issue 6, Volume 4, June 2008.

[11] G. G. Rajput, Rajeswari Horakeri, Sidramappa Chandrakant- Printed and Handwritten Mixed Kannada Numerals Recognition Using SVM- IJCSE- Vol. 02, No. 05, 2010, 1622-1626

[12] N.Arica and F.T.Yarman_Vural-One dimensional representation of two dimensional information for HMM based handwriting recognition, Pattern recognition letters, vol-21(2000)583-592