Kannada Handwritten Digits Recognition: Neural Network Approach

Vishweshwrayya C. Hallur[1], Dr. R. S. Hegadi[2]

[1]Research Scholar, Department of Computer Science,
Bharathiar University, Coimbatore-641046, Tamilnadu
vishuhallur@gmail.com

[2]Asociate Professor and Head, School of Computer Science,
Solapur University, Solapur
rshegadi@gmail.com

Abstract: ***In this paper, our task is to design and train a high efficient, accuracy Back Propagate Neural Network, in order to recognize the Kannada handwritten numbers from 0 to 9. Before we pass the data to the BP Neural Network, we have to use image processing to achieve the good accuracy. There are two major components constituting the module, image preprocessing and recognition. The preprocessing component consists of noise removal, binarization, skew detection, segmentation ect based on the type of the image. Here back-propagation networks to hand-written Kannada numeral recognition. Architecture of the network is extremely forced and specifically designed for recognition of hand written Kannada numeral recognition task. The input to the network consists of normalized binary images of isolated numerals. The method has achieved accuracy of 95%.***

Keywords: OCR, Back Propagation Neural Network, Kannada Handwritten Numeral Recognition..

1. Introduction

The conversion of handwritten kannada Numerals on a paper into digital (electronic) form facing more problems with great electronic importance. There are some solutions for clearly printed documents in many language scripts. Solutions to hand writing still face many challenges in almost all the languages. That is due to styles differs from person to person, due to sizes, shapes, etc. Here for kannada handwritten numeral recognition, the ideas were taken from various papers of different language handwritten recognition [1, 2].

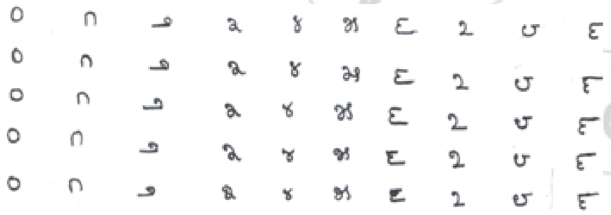


Figure 1: Sample Handwritten Kannad Numerlas from 5 people

From the view point of recognition, kannada numerals consist of exactly ten digits of which only a digit (0) is commonly used in Kannada and English. The numbers are written horizontally from left to right like English called sample.

Recognition of handwritten Kannada digit is particularly challenging task due to more curves in shapes. Furthermore the digits are usually not written in print fashion. Segmentation of digits from numbers is often easier as they are not connected. Furthermore the high individual variation in writing kannada digits contributes to the problem. Many techniques have been developed for identifying individual digits. While various algorithms vary in their implementation but most follow a typical path from document to UNICODE text.

೦	೧	೨	೩	೪	೫	೬	೭	೮	೯
0	1	2	3	4	5	6	7	8	9

Figure 2: Kannada digits and their equivalent English digits

2. Preprocessing

Scanned image is converted into a form suitable for later steps pre-processing stage. Pre-processing step involves noise removal, smoothing, binarization, segmentation, normalization of characters, slant adjustments, etc [3].

2.1 Noise Reduction

In Matlab, there are some function for removing noise, such as wiener2, medfilt2 and ordfilt2. Wiener2 low pass filters an intensity image that has been degraded by constant power additive noise. Medfilt2 performs median filtering of the image matrix in 2D. Each output pixel contains the median value in the two dimensions neighborhood around the corresponding pixel in the input image. Ordfilt2 perform 2-D order-statistic filtering, is the extension of medfilt2. Here is the example using medifilt2 to remove the noise from the image.

Figure 3: Gray color Handwritten image of Digit ’4’.

2.2 Binarisation

Considering a gray scale image, it is first binarized. Since some information will somehow be lost during binarization, a variable thresholding technique could be used in Matlab. Although variable thresholding cannot completely compensate for the information loss mentioned above, it at least preserves information that may be lost when using a constant binarization method. There are two purposes for the binarization step: highlighting characters and suppressing background. However, both desired and undesired image areas often appear during binarization. Matlab provided lots of function to convert image to different formats.

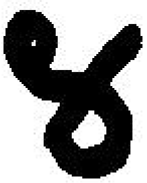


Figure 4: Binary Image of Handwritten Kannada Digit '4'

2.3 Normalization

Normalization provides a enormous reduction in data size, thinning extracts the contour information of the digits. Algorithms for thinning are frequently considered for converting offline handwriting to somewhat similar to online handwriting. However, thinning algorithms would introduce artifacts such as spurs.

2.4 Skew Correction

These methods are used to adjust the paper document with the co-ordinate system of the scanner. The approaches for skew detection include correlation, projection profiles and Hough transform.

2.5 Slant Removal

The slant of the handwritten numerals differs from user to user. Slant removal techniques are used to normalize all the numeral characters to a standard form by popular slanting correcting technique is called computation of the average angle of near vertical elements.

2.6 Digit Segmentation

After the binarization, we should get rid of the white areas which do not affect the recognition. First: we need to find out inverse of the image that is by making 0 to 1 and 1 to 0 in the binary image as shown in the below image.

Second: From the new image, find out the edges of the image Third: Find out the black area above and under the digit, cut them off as well.

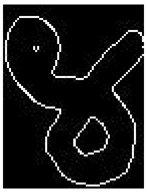
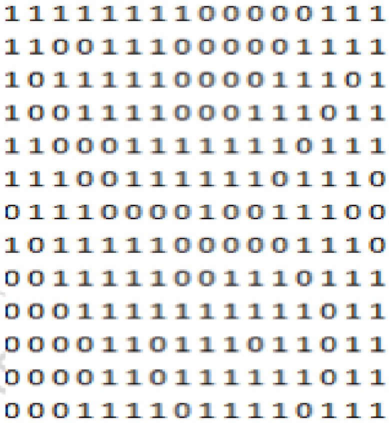


Figure 5: Complement Image of Handwritten Kannada Digit '4'

2.7 Resizing

The numbers are resized using the nearest neighbor method. The area of the original number is reduced if noisy edges consisting of single pixels without neighbors are found. This makes the system shift invariant. The numbers are taken from the image and placed into blocks of 16 pixels wide by 16 pixels high in a binarised form. The size is chosen as a compromise between good resolution and simplicity and is easily divisible for template matching [4].



```
1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1
1 1 0 0 1 1 1 0 0 0 0 0 1 1 1 1
1 0 1 1 1 1 1 0 0 0 0 1 1 1 0 1
1 0 0 1 1 1 1 0 0 0 1 1 1 0 1 1
1 1 0 0 0 1 1 1 1 1 1 1 0 1 1 1
1 1 1 0 0 1 1 1 1 1 1 0 1 1 1 0
0 1 1 1 0 0 0 0 1 0 0 1 1 1 0 0
1 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0
0 0 1 1 1 1 1 0 0 1 1 1 0 1 1 1
0 0 0 1 1 1 1 1 1 1 1 1 1 0 1 1
0 0 0 0 1 1 0 1 1 1 0 1 1 0 1 1
0 0 0 0 1 1 0 1 1 1 1 1 1 0 1 1
0 0 0 1 1 1 1 0 1 1 1 1 0 1 1 1
```

Figure 6: The matrix form of 16X16 binary image of digit 4

3. Proposed System

The system uses a feed forward neural network consisting of three layers. The input layer comprises 256 neurons, one of each pixel of the binary character image (16*16). The hidden layer has 25 neurons. The value is calculated by experimentation, beginning with heuristic values and adjusting them until the greatest accuracy was achieved. The number of neurons in the output is 1, because we need to recognize one number a time. The hidden layer used the Tan Sigmoid method, which allows for non-linearity. The output layer used Pureline method. The performance function used MSE. The training function used traincgf. The structure showing below:

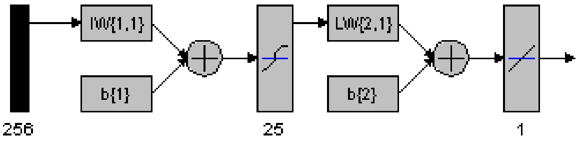
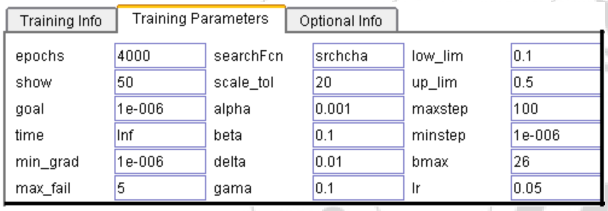


Figure 7: Digit Recognition Neural Network structure

Training and Testing the Neural Network: A neural network is trained by adjusting the weights and biases that are used in the feed forward calculations, which perform the digit recognition. The neural networks were trained by propagating the inputs forward through the network, then propagating the sensitivities backward through the network and updating the weights and biases using resilient back propagation. One number we are using 4 different handwritten fonts and different sizes in training the neural networks [5]. Properly trained back propagation networks tend to give reasonable answers when presented with inputs that they have never seen. Generally, a new input leads to an output similar to the correct output for input vectors used in training that are similar to the new input being presented. Sigmoid functions are characterized by the fact that their slope must approach zero as the input gets large. This causes a trouble when using steepest descent to train a multilayer network with sigmoid functions, since the gradient can have a small magnitude, and hence, cause small changes in the weights and biases, even though the weights and biases are far from their optimal values. Instead the resilient back propagation is used to eliminate these harmful effects of the magnitudes of partial derivatives. Only sign of the derivative is used to determine the direction of the weight update, the magnitude of the derivative has no effect on the weight update [6]. The size of the weight change is determined by a separate update value. This method is accurate, quick and does not require a lot of computer memory, which makes it feasible for this application. The performance goals for the training were set up using the recommendations of previous researchers. The performance goals are shown in below table.

4. Results

A set of 50 images was used to test the system, included among these were images have noise, different sizes, colors, bold, incline and fonts. The results showed us, this digit recognition system based on BP Neural Networks can achieve accuracy around 95% in order to get the perfect results; we use the round function to get the nearest integers of the results.



epochs	4000	searchFcn	srchcha	low_lim	0.1
show	50	scale_tol	20	up_lim	0.5
goal	1e-006	alpha	0.001	maxstep	100
time	Inf	beta	0.1	minstep	1e-006
min_grad	1e-006	delta	0.01	bmax	26
max_fail	5	gama	0.1	lr	0.05

Figure 8: Handwritten Kannada Digits Recognition Training Parameters

References

[1] A. F. R, Rahman, and M. C. Fairhurst, ‘Recognition Of Handwritten Bengali Characters: A Novel Multistage Approach,” Pattern Recognition, Vol. 35 pp. 997–1006, 2002.

[2] R. Chandrashekaran, M. Chandrashekaran and Gift Siromaney, “Computer Recognition Of Tamil, Malayalam And Devanagari Characters,” Journal Of Iete, Vol. 30, No. 6, 1984

[3] Vishweshwarayya C. Hallur, Dr. R. S. Hegadi, “Offline Kannada Hand-written Numeral Recognition: Holistic Approach,” International Conference On Emerging Research In Computing, Information, Communications And Applications, Elsevier Publications, 2014, P. 632-637.

[4] Vishweshwarayya C. Hallur, Dr. R. S. Hegadi, “Template matching approach for Handwritten Kannada Numeral Recognition,” International conference on Recent Trends in Computing, IJCA, 2012, pp. 11-12.

[5] Jinfeng Bai, Zhineng Chen, Bailan Feng, BoXu, “Image Character Recognition Using Deep Convolutional Neural Network Learned From Different Languages,” ICIP, 2014, Pp. 2560-2564.

[6] Robert P. W. Duin, “Superlearning and Neural Network Magic,” Pattern recognition Letters, 1994, North-Holland, pp. 215-217.