DEVANAGARI CHARACTER RECOGNITION USING FEATURE SET SELECTION ALGORITHM

Anita Venugopal[1], Mukesh Madanan[2], Akhil[3]

Abstract

Presently these days recognizing the handwritten character recognition is getting high essentialness in light of various applications like educational field, digitized signature verification, bank processing, postal code acknowledgment, electronic library and so on. exceptionally less work is ac-counted in the research of Devanagari hand written character acknowledgment, so that there is an enormous extent of research right now. Some potential challenges adding to the unlawful execution of different frameworks for seeing deciphered characters are: various shapes, broken characters, various tendencies and measures, and so on. To overcome these types of issues, initially, we introduce a pixel intensity histogram based feature for the special character recognition, it identifies the special symbols and characters and different types of characters. Further, we used selection process done by improved rule based feature set selection algorithm. Dataset is collected and with help of this proposed improved rule based feature set selection algorithm, the accuracy of character identification is improved. Further, we use recurrent-artificial neural network classifier for classification and recognition process to classify different types of handwritten characters. The performance of proposed model is compared with the existing designs in terms of higher accuracy and speed in classification and recognition.

Keywords: *artificial neural network, a pixel intensity, improved rule based feature set selection algorithm, devanagari hand written character*

Introduction

Over couple of years, profound learning approaches have been effectively applied to different regions, for example, picture arrangement, discourse acknowledgment, malignancy cell recognition, video search, face location, satellite symbolism, perceiving traffic signs and so on. Character acknowledgment is one of the regions where AI strategies have been broadly tested.

Deep learning technology improves the ability of machines to learn on its own without being told how to do. Some networks may use best case scenario information for low to high level functions and may reduce the gradient degradation problem caused by lower levels [1]. Structure of advanced learning with Recurrent Neural Network (RNN), Convolutional Neural Network (CNN) and Long short-term memory(LSTM) gives machines

[1] Dept. of FP IT Unit, Dhofar University, Salalah, Oman, Email: anita@du.edu.com

[2] Dept. of Computer Science, Dhofar University, Salalah, Oman, Email: mukesh@du.edu.com

[3] Dept. of Industrial Engineering, Concordia University, Montreal, Concordia, Email: akhilvg96@gmail.com

Received: 26 Apr 2020 | Revised: 24 May 2020 | Accepted: 28 Jun 2020

the ability to learn from previous results. Long short-term memory (LSTM) is a special type of recurrent neural network (RNN), designed to store results for longer duration [2]. It can connect backwards and can solve problems related to cursive writings. The noise level does not effectively restore different images from deep CNN training data with limited noise levels in the training data [3]. In particular, the best performing networks for sentiment analysis and data retrieval are the Conventional Neural Network (CNN) [4].

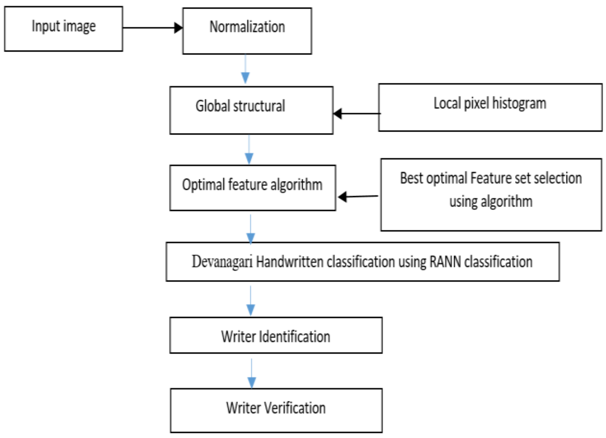

Fig. 1. Block diagram of proposed system model

PROPOSED HANDWRITTEN DEVANAGARI CHARACTER DETECTION TECHNIQUE

K-Nearest Neighbor Algorithm

The k-nearest-neighborhood (k-NN) algorithm uses non parametric approach for classification. This algorithm takes input which consists training examples which are closer to the feature space. An object is classified by most of its neighbors, and the object belongs to the most general category of its nearest neighbors, k (k is a positive integer, usually small). If k = 1, the object is assigned to the nearest neighbor class. The idea of algorithm, K for the nearest neighbor is simple. To classify a new role, the system searches your nearest neighbor on training records and uses your closest neighbors in the plane. The K-NN algorithm can be described using the following equation:

$$y(d_i) = \arg\max_K \sum_{xj \in KNN} sim\left(d_i, x_j\right) y\left(x_j, c_k\right)$$

Where, di is a test character, xj is a neighbor of the training group, denotes that y (xj, ck) ∈{0, 1} belongs to class ck, sim (di, xj).) Is the similarity function of D. The performance of this algorithm is strongly dependent on two factors, namely the corresponding similarity function and the appropriate value for the k parameter. The similarity function is the Euclidean distance. This is given by the following equation:

$$f\left(x, p^2\right) = \sum_{i=1}^{N} \left(x_i - p^2{}_i\right)$$

Local Binary Pattern

The local binary model (LBP) is mainly found for shape classification. In this method, a small picture window and the difference in density between the central pixel and its neighbour's N is located within the circumference of the radius R circle and is encoded in binary format. Depending on the difference in intensity, a binary value (0 or 1) is assigned to each pixel of the surrounding pixels, which is multiplied again by a certain number of weights and collected to produce the result. This value is replaced by the value of the central pixel, which is the binary sample of the central pixel. The entity vector is calculated by creating a graph of this local bitmap. Here, N and R denote the number and radius of adjacent pixels, respectively.

$$LBP(N,R) = \sum_{i=1}^{N} 2^{i-1} XD_1(I_i, I_c)$$

Feature set selection algorithm to select best optimal feature

Central cluster begin by estimating the initial mass weight. Mass is the cumulative probability of the intensity of pixels in a block. When the input image is displayed with a density of 0 pixels (0, 1, 2. ... L - 1), the number of pixels in the density level and the total number of pixels $f = f0 + f1 + \cdots + fL - 1$. For the given pixel density, the intensity (i) is given the probability of occurrence:

$$P(i) = \frac{f_i}{f},\ P(i) \geq 0,\ \sum_{i=0}^{L-1} p(i) = 1$$

In this work, the variable is understood as the cluster, which is the number associated with the pixel density with its group. This indicates that two or more image pixels belonging to the same group can easily be obtained for the same group designation for group purposes. The index t of the pixel intensity, is as follows:

$$t = (\text{int})\left(\frac{J*Q}{K}\right), Q, K = 2,3,\ldots.L$$

cluster center $\mu_i = \frac{\varpi'_j}{\varpi_j}$

PICA time complexity analysis can be performed for each step of the algorithm.

Algorithm 1

Input: $M \times N$grayscale image, K number of clusters.

Output: M×Ngrayscale image.

1	Let S = ($e0$, $e1$, . . . ,−1), $Q \leq L$represents the set of image pixel intensities and assuming BCV is the between cluster variance.
2	**for all** j = 0, 1, . . . , − 1 **do**
3	t = (int) ((j∗Q)/K)
4	estimate the jth cluster centroid using the pixel intensity et
5	assign cluster label $j$to the pixel intensity et
6	swap the pixel intensity et with the pixel intensity ej
7	**end for**
8	**for all** i = K,K+ 1, . . . , Q − 1 **do**
9	**for all** j = 0, 1, . . . ,K − 1 **do**
10	tentatively allocate pixel intensity ei to the jth cluster
11	BCV = wj∗(variance of the jth cluster centroid)
12	**for all** k = 0, 1, . . . ,K − 1 **and** $j$not equal to k**do**
13	BCV = BCV + wk∗(variance of the kth cluster)
14	**end for**
15	**end for**
16	permanently allocate ei to the cluster that gives maximum BCV
17	assign cluster label to the allocated pixel intensity ei
18	update cluster centroid
19	**end for**
20	compute output image
21	**stop**

International Journal of Psychosocial Rehabilitation, Vol. 24, Issue 07, 2020
ISSN: 1475-7192

Recurrent neural network (RNN)

Recurrent neural network is a special branch of artificial intelligence which is used to solve handwriting recognition problems and image identification problems. We used a dataset consisting of 2000 samples to find the Devanagari alphabet. It is necessary to calculate the training for the entire RNN with random initialization and this may not be possible due to the limited data set. Typically, pre-training models are trained on this network.

Following are mainly two techniques that are used transfer learning:

- Fine tuning: In randomly configuring different layer weights, it begins with a grid containing a grid, and the rest of the training is performed using the new data set as usual.

- Fixed Entity: In this process, weights of the various convolutional layers are frozen, which act as an entity extractor, but the weights of the connected layer are completely updated.

In some complex tasks, a fully connected layer can be formed, as well as some of the up-per convolutional layers. In this paper, we used Convent as a static operational extract.

DATA PREPARATION

Devanagari's handwritten alphabetical dataset contains 46 categories with 2000 images for each category. We divided a data set of group consisting 32 x 32 photos in grayscale. Pixel filler is used for images. The numeric difference in a handwritten dataset created by different people in handwritten documents

Fig. 2. Sample image of handwritten characters

I. THE PERFORMANCE OF CONVOLUTIONAL NEURAL NETWORK

CNNs accept input, weight, bias values and with these they are able to recognise characters. CNNs convert characters to a form that are easy for processing. It consists of layers like input, convolution and pooling.
CNNs use spatial local bonding, which creates a pattern of local relations between neurons..

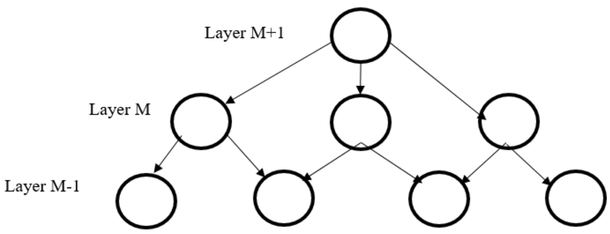

Fig. 3. CNN network process

Convolutional layer: two-dimensional filtering between the input images x and the filter bank w, to generate another image h. CT connection table represents input and output correspondence, and candidate responses to inputs connected to one out-put image are linearly coupled. Each line of CT scan has the following meanings: (inputImage, filterId, outputImage).

In addition, each neuron has an offset. The model determines the probability distribution.

Received: 26 Apr 2020 | Revised: 24 May 2020 | Accepted: 28 Jun 2020

$$P(x,h) = \frac{e^{-E(x,h)}}{Z}$$

The energy function is defined as the separation function Z with E (x, h).

$$E(x,h) = -b'x - c'h - h'Wx$$

$$Z(x,h) = \sum_{x,h} e^{-E(x,h)}$$

FEATURE EXTRACTION METHOD

The method of obtaining gradient direction properties is more popular because it provides better performance than another method. After discovering the gradient image (surrounding pixels), there are two different ways to estimate the object vector. The first method analyzes the gradient image in the 8x8 grid and sets the gradient size for each area in each area of the grid. Another method uses a Gaussian mask or filter in the grid area to find the entity vector. The same characteristics are recorded in the work. The modular image is divided into four images, each consisting of one dimensional pixels. Each image is filtered with a 16 x 16 mask and scanned into an 8 x 8 grid. A fully functional vector has 256 dimensions (8 * 8 * 4).

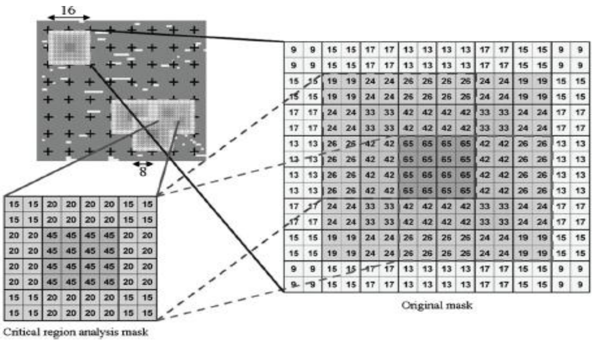

Fig. 4. Mask and entity extraction process

TRAINING AND TESTING

Data was tested for 7 basic data sets previously obtained in 6 training groups and 1 test group. This parameter is only used to assess the performance of functionality-based recognition algorithms, since these groups have no particular limitations and therefore do not require structural identification.

User #	Sequential (CORE7)	Frequently Occurring Characters (FREQ)	Randomly Chosen Characters (RAND)	Complete set (FREQ+RAND)
0	91.21%	92.12%	79.01%	89.18%
1	92.24%	92.91%	81.05%	90.12%
2	93.16%	93.68%	83.10%	91.05%
3	94.39%	94.56%	85.56%	92.67%
4	95.51%	95.11%	87.65%	93.57%
5	96.19%	96.34%	89.65%	94.32%
6	97.35%	97.12%	91.34%	95.65%
7	98.46%	98.43%	93.56%	97.01%
8	99.59%	99.14%	95.89%	98.76%
Average	98.27%	98.45%	93.34%	91.01%

The accuracy of the percentage of correctly identified test samples is given in the total number of test samples of the respective test group.

The CORE7 test suite used for consecutive tests contains 441 characters. The most frequently used character set (FREQ) is 200 characters. There are also 250 test characters in RNN. Table 1 shows the posterior effect of the basic classification and the detection rate of 98.9% for the database obtained.

The inclusion of additional highlight points for letters of comparative form has the potential to improve classification error and accuracy by 4.63% (some examples).

However, the productivity of the proposed strategy and the current trading system is estimated using different datasets of Devanagari terms.

S.No	Features	Parameters	Accuracy
1	Chain code Regular expression & MED	Characters	82%
2	Chain code Quadratic	Characters	80.36%
3	Structural approach	Characters	89.12%
4	Shadow & CH MLP & MED	Characters	90.74%
5	Vector distance Fuzzy sets	Characters	90.65%
6	Gradient	Characters	94.1%
7	Eigen deformation Elastic matching	Characters	94.91%
8	Gradient SVM & MQDF	Characters	95.13%
9	Gradient & Gaussian filter Quadratic	Characters	94.24%

TABLE I. RESULT ANALYSIS

ACKNOWLEDGMENT

We are very much thankful to our organization for providing all kind of logistic and other supports.

REFERENCES

[1] Jung, E., Chikontwe, P., Zong, X., Lin, W., Shen, D. and Park, S.H., 2019. Enhancement of perivascular spaces using densely connected deep convolutional neural network. IEEE Access, 7, pp.18382-18391.

[2] Chen, J., Yang, R., Zhang, C., Zhang, L. and Zhang, Q., 2019. DeepGly: A Deep Learning Framework With Recurrent and Convolutional Neural Networks to Identify Protein Glycation Sites From Imbalanced Data. IEEE Access, 7, pp.142368-142378.

[3] Isogawa, K., Ida, T., Shiodera, T. and Takeguchi, T., 2017. Deep shrinkage convolu-tional neural network for adaptive noise reduction. IEEE Signal Processing Letters, 25(2), pp.224-228

[4] Alali, M., Sharef, N.M., Murad, M.A.A., Hamdan, H. and Husin, N.A., 2019. Narrow Convolutional Neural Network for Arabic Dialects Polarity Classification. IEEE Access, 7, pp.96272-96283