SLOPE AND CURVE CODING TO RECOGNIZE ENGLISH NUMERALS

Binod Kumar Prasad

Guru Nanak Institutions Technical Campus, Ibrahimpatnam, T.S., Hyderabad, India.

Email: binod.krpr@gmail.com

Article History: Received on 12th April 2020, Revised on 27th May 2020, Published on 16th June 2020

Abstract

Purpose: Lines and Curves are important parts of characters in any script. Features based on lines and curves go a long way to characterize an individual character as well as differentiate similar-looking characters. The present paper proposes an English numerals recognition system using feature elements obtained from the novel and efficient coding of the curves and local slopes. The purpose of this paper is to recognize English numerals efficiently to develop a reliable Optical Character recognition system.

Methodology: K-Nearest Neighbour classification technique has been implemented on a global database MNIST to get an overall recognition accuracy rate of 96.7 %, which is competitive to other reported works in literature. Distance features and slope features are extracted from pre-processed images. The feature elements from training images are used to train K-Nearest-Neighbour classifier and those from test images have been used to classify them.

Main Findings: The findings of the current paper can be used in Optical Character Recognition (OCR) of alphanumeric characters of any language, automatic reading of amount on bank cheque, address written on envelopes, etc.

Implications: Due to the similarity in structures of some numerals like 2, 3, and 8, the system produces respectively lower recognition accuracy rates for them.

Novelty: The ways of finding distance and slope features to differentiate the curves in the structure of English Numerals is the novelty of this work.

Keywords: *Slope Coding, Curve Coding, English Numerals Recognition, K-Nearest Neighbour, Optical Character Recognition, Delta Distance Coding.*

INTRODUCTION

Machine learning has been a dynamic field with rapid evolutions aimed at developing and improving algorithms that allow computers to learn from data and recognize patterns. Old documents are written on papers. These papers are getting tarnished day-by-day due to the aging effect. Therefore, writing on these papers are not safe for a longer time. We are bound to lose a lot of valuable information unless the old writings are typed into computers. The manual process of typing is time and resource-consuming in terms of cost. Again, several typo errors will provide misleading information. Here lies the importance of Optical Character Recognition (OCR). It makes machines able to recognize any text automatically making archiving and retrieving old documents very easy. At present, OCR is helping in various applications like mail sorting, bank cheque processing, and reading machines for the blind, automatic number plate recognition. The motivation behind the present paper is to develop an efficient Optical Character Recognition system to help society to preserve ancient knowledge.

LITERATURE REVIEW

The off-line numeral recognition system consists of the process of classification among ten different classes. The uniqueness of one's handwriting makes it rather more challenging to develop an off-line recognition system with maximum efficiency. Different ways of thinking towards the solution of the above problem reports various techniques ([Liu, C.L et al., \(2004\)](#), [Karungaru et al., \(2013\)](#)), have put forward a neural network-based character classification system. Features have been derived from star layered histograms. Centre of gravity (CoG) is determined from the contour of character region and several lines at an equal distance are considered from CoG to the contour. The first point of intersection of the line with character gives rise to the first layer of the histogram and so on unless the line touches the boundary of the region containing the character. The system shows an overall accuracy of 97.1% on the MNIST database. [Qacimy, B. et al., \(2014\)](#), have tried to improve the discriminatory ability of digit classification system by means of DCT upper left corner coefficients (225), DCT zig-zag coefficients (100), Block based DCT ULC coefficients (256) and Block based DCT zig-zag coefficients(160). Classification is done by means of SVM and obtained global accuracy on the MNIST dataset are 98.66%, 98.71%, 98.73% and 98.76% respectively. [Bhattacharya, U. & Chaudhuri, B.B. \(2009\)](#), have proposed a multistage recognition system in the cascaded form to recognize mixed digits of Devanagari and Bangla under Indian script and English. Wavelet-based multi-resolution representations and multilayer perceptron (MLP) are utilized for the purpose. They have presented two large databases for the above Indian scripts whereas for English numerals, the MNIST database has been preferred. The system provides misclassification rates of 0.81% and 1.01% on training and test MNIST English samples respectively to get overall accuracy of 98.79%. [Celar, S. et al., \(2015\)](#), have used structural characteristics as global features and local descriptors based on Scale-Invariant Feature Transform (SIFT) which is invariant to scaling, rotation and even affine transformation to recognize digits in student ID. These features are organized by the Bag of Words (BoW) model. They tested the MNIST test database with

MLP to achieve a correct recognition rate of 94.76%. [LeCun, Y. et al., \(1998\)](#), have reviewed several methods of HCR and found CNN with a Gradient-based learning method to be superior as it can alleviate the variability of 2D shapes. GTN also uses CNN along with global training techniques to yield promising results. [Chen, G. et al., \(2016\)](#), have combined Conditional Random Fields with deep learning to recognize handwritten words. Deep features are learned and sequences are labeled in a unified framework. Deep structure is pre-trained with stacked restricted Boltzmann machines and optimization of the entire network has been done with an online learning algorithm. [Xiao, X. & Suen, C. \(2012\)](#), have presented a hybrid classifier consisting of Convolution Neural Network (CNN) and Support Vector Machine (SVM) to recognize handwritten digits. In order to extract features, CNN has been applied whereas recognition is done with SVM. [Lopes, G. et al., \(2016\)](#), have extracted features from the signature of characters and applied Optimum Path Forest (OPF) classifier for recognition of digits. They have got satisfactory results with a respectively lesser time of processing. The research work ([Aziz, T. et al., \(2018\)](#)) proposes a directional pattern approach for feature extraction of Bangla numeric characters. It uses Local Directional Pattern (LDP) and Gradient Directional Pattern (GDP) for feature extraction and then two machine learning algorithms, K-Nearest Neighbor (KNN) and Support Vector Machine (SVM), to classify the numeric characters. The results on the benchmark dataset (CMATERdb 3.1.1) demonstrate an astounding recognition rate of accuracy of 95.62% without preprocessing the data. [Ashiquzzaman, A. & Tushar, A. K. \(2017\)](#), have presented a novel algorithm based on deep learning neural networks using appropriate activation function and regularization layer. The experiment provides significantly improved accuracy compared to the existing Arabic numeral recognition methods. The proposed model gives 97.4 percent accuracy, which the authors claim to be the recorded highest accuracy of the dataset used in the experiment. However, the present paper puts forward an efficient automatic offline recognition system for handwritten English numerals with an acceptable accuracy of 96.7%.

METHODOLOGY

The methodology of the proposed work consists of the following processes carried out sequentially-

1. Pre-processing

The samples are resized to a common size of 50×50 keeping the aspect ratio intact so that the same Matlab code for feature extraction and classification could be applied. Thresholding has been done by means of Otsu's method that binarizes the gray level images. Skeletonisation is done to obtain the single-pixel thickness of the numeral images. It is preceded by Thickening so that no part of an image could disappear because of the thinning operation of Skeletonisation. Thickening includes the process of dilation which bridges any unintentional discontinuity (gap) while writing any character.

2. Feature extraction

Features are aimed at coded representation of a curve. They have been classified broadly into two types based on the parameter of coding namely Distance Features and Slope Features.

Distance Features (DF)

The paper adopts Delta Distance Coding (DDC) to extract Distance Features which is basically, a differential approach towards coding a curve based on distance. In the differential approach of coding, the difference between magnitudes (pixel intensities in case of images) of two successive samples of a curve is considered that is absolutely lesser than either of the magnitudes. It reduces the complexity of the system. Delta Distance Coding (DDC) has been used to differentiate concavity from convexity while encoding the local curvature in a numeral sample. DDC has been applied both vertically and horizontally.

Vertical Delta Distance Coding (VDDC)

The pre-processed image of size 50×50 is divided into four equal sub-images each of size 25×25 to get local features. Twenty-five samples (a, b, c, d,...) of curves in each of the four sub-image spaces are considered and the distance of the samples from reference level (pa, qb, rc, sd,) is noted. The upper edge of the sub-image is conventionally taken as a reference level. Delta Distance code for one sample is written on the basis of comparison between current distance and that of the just previous sample.

Figure 1: Scheme of Vertical Delta Distance Coding (VDDC)

Coding logic – Let us consider,

D_c = distance of current sample 'c' from the reference level;

D_b = distance of just previous (on left side) sample 'b' from the reference level;

If (i) $D_c < D_b$; code will be '1'

(ii) $D_c > D_b$; code will be '2'

(iii) $D_c = D_b$; code will be '0'

The coding for the curve in Figure 1(a) is '11022' whereas for that in Figure 1(b), it is '22011'. Two different codes for concave and convex curves assure that VDDC is able to distinguish the vertical concavity and convexity. Total feature elements out of this feature are equal to $25 \times 4 = 100$.

$$F(VDDC) = 100$$

Horizontal Delta Distance Coding (HDDC)

Twenty-five samples are considered from each of four sub-images of size 25×25 corresponding to a sample image. The left edge is conventionally considered as reference level as shown in Figure 2. The coding logic in HDDC is the same as that followed in VDDC.

Figure 2: Sample points with corresponding distances in Horizontal Delta Distance Coding

HDDC for curve in Figure 2(a) is '11022' and that for the curve in Figure 2(b) is '22011'. So, HDDC can well differentiate horizontal convexity and concavity. Total feature elements out of HDDC = $25 \times 4 = 100$ i.e., $F(HDDC) = 100$.

Thus, $F(DF) = F(VDDC) + F(HDDC) = 100 + 100 = 200$.

Slope Feature (SF)

Slope Feature represents variations in slope. It is obvious from Fig. 3(a,b) that a concave curve with an opening along the X-axis has gradually increasing slope going along the curve in the direction of arrow mark whereas a convex curve has a decreasing slope moving down the curve. In the same way, a concave curve with an opening along Y-axis has a decreasing slope while a convex curve possesses increasing slope with respect to Y-axis. In the current paper, SF is extracted from the feature elements of VDDC and HDDC. Any two successive 1's in VDDC or HDDC reflects increasing (positive) local slope and it is coded as a positive slope code '3'. On the other hand, two successive 2's in DF reflects decreasing (negative) local slope and it is coded as negative slope code '4'. For DF code '11022' and '22011', SF code will be '3004' and '4003' respectively.

Figure 3: Slope change in curves with axis along X-axis

The local slope of a curve will be coded as per the scheme shown in Figure 3. Therefore, Feature elements out of Slope Feature i.e.

$$F(SF) = [24 \times 4] = 96.$$

FINDINGS / RESULTS

K-Nearest Neighbor classifier has been trained with both Distance features and Slope features together and MNIST test data are classified with parameter 'K' equals 1 and 3. KNN classifier applies the Euclidean distance metric to locate the nearest neighbor and Majority rule with nearest point tie-break for classification. Table 1 shows the individual rate of recognition of all the ten numerals. The average rate of recognition of the system is 96.7%.

Table 1: Rates of recognition of digits

Digits	KNN with	
	K=1	K=3
0	96.8	97.7
1	98.1	98.1
2	96.2	96.2
3	96.7	90.9
4	95.2	98.0
5	97.6	97.6
6	97.5	98.1
7	97.2	96.1
8	86.0	89.3
9	98.1	95.7

Figure 4: Line diagram showing individual results using KNN with K=1, 3

DISCUSSION / ANALYSIS

In order to compare the method and results of the current paper with those of existing work, the reference paper [Babu et al. \(2014\)](#) has been considered. They have classified Arabic numerals from the MNIST database with KNN classifier and derived features from the number of holes in an image, presence of water reservoirs and fill-hole density. The system reports an overall accuracy of 96.94%. Individual percent recognition rates of all the digits in the two systems have been tabulated in Table 2 and plotted in Figure 5. For the current paper, the better rate of recognition out of those with K=1 and K=3 in Table 1 has been considered for comparison in Table 2.

Table 2: Comparison of individual accuracy

Digits	Percent Accuracy in paper	
	U.Babu et al. [12]	Current Paper
0	96.47	97.7
1	97.95	98.1
2	97.26	96.2
3	97.56	96.7
4	97.27	98.0
5	97.47	97.6
6	96.09	98.1
7	95.72	97.2

8	95.71	89.3
9	97.23	98.1

Figure 5: Columns showing recognition accuracy of individual digits in the two systems

CONCLUSION

The current work presents an English numerals recognition system with features derived from different ways of curves and slope coding and K-Nearest Neighbour as a recognition tool. The system yields an overall recognition accuracy of 96.7% which justifies the efficiency of the features. The proposed work has been able to differentiate lines and curves and ultimately it helped to avoid any ambiguity while doing classification. Thus, the paper has been able to achieve its aim.

LIMITATION AND STUDY FORWARD

It is obvious from Table 1 that some numerals like '2', '3', and '8' have shown respectively lower recognition rates. It is because of the similarity among their structures. The future work will be directed towards finding some demarcating features to distinguish these numerals so that their individual and also average recognition accuracy of the whole system could be improved.

ACKNOWLEDGEMENT

Thanks to the authority of Guru Nanak Institutions Technical Campus, Ibrahimpatnam, Hyderabad, India to provide with internet and other allied facilities to carry out research works.

REFERENCES

1. Ashiquzzaman, A. & Tushar, A. K. (2017). Handwritten Arabic numeral recognition using deep learning neural networks. *International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*, Dhaka, Bangladesh, IEEE. <https://doi.org/10.1109/ICIVPR.2017.7890866>
2. Aziz, T., Rubel A.S., Salekin. S. & Kushol, R. (2018). Bangla handwritten numeral character recognition using directional pattern. 20th *International Conference of Computer and Information Technology (ICCIT)*, Dhaka, Bangladesh, IEEE. <https://doi.org/10.1109/ICCITECHN.2017.8281820>
3. Babu, U., Venkateswarlu, Y. & Chintha, A. (2014). Handwritten Digit Recognition Using K-Nearest Neighbour Classifier. *Proc. of World Congress on Computing and Communication Technologies*, IEEE. <https://doi.org/10.1109/WCCCT.2014.7>
4. Bhattacharya, U. & Chaudhuri, B.B. (2009). Handwritten Numeral Databases of Indian Scripts and Multistage Recognition of Mixed Numerals. *IEEE transactions on pattern analysis and machine intelligence*, 31(3), 444 - 457. <https://doi.org/10.1109/TPAMI.2008.88>
5. Celar, S., Stojkic, Z. & Seremet, Z. (2015). Classification of Test Documents Based on Handwritten Student ID's Characteristics. 25th *International Symposium on Intelligent Manufacturing and Automation (DAAAM)*, Procedia Engineering (Elsevier), 782 – 790. <https://doi.org/10.1016/j.proeng.2015.01.432>
6. Chen, G., Li, Y. & Srihari, S. (2016). Word recognition with deep conditional random fields. *International Conference on Image Processing (ICIP)*, IEEE. <https://doi.org/10.1109/ICIP.2016.7532694>
7. Karungaru, S., Terada, K. & Fukumi, M. (2013). Hand Written Character Recognition using Star-Layered Histogram Features. *SICE Annual Conference*, Nagoya University, Japan, 1151-1155.
8. LeCun, Y., Bottou, L. & Bengio, Y. (1998). Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE*, 86(11), 2278-2324. <https://doi.org/10.1109/5.726791>
9. Liu, C.L., Nakashima, K., Sako, H. & Fujisawa, H. (2004). Handwritten Digit Recognition: Investigation of

- Normalization and Feature Extraction Techniques. *Pattern Recognition* 37(2), 265-279. [https://doi.org/10.1016/S0031-3203\(03\)00224-3](https://doi.org/10.1016/S0031-3203(03)00224-3)
10. Lopes, G., Silva, D., Rodrigues, A. & Filho, P. (2016). Recognition of handwritten digits using the signature features and Optimum Path Forest classifier. *IEEE Latin America Transactions*, 14, 2455-2460. <https://doi.org/10.1109/TLA.2016.7530445>
 11. Qacimy, B., Kerroum, M. & Hammouch, A. (2014). Feature extraction based on DCT for handwritten digit recognition. *International Journal of Computer Science*, 11(2), 27-33.
 12. Xiao, X. & Suen, C. (2012). A novel hybrid CNN-SVM classifier for recognizing handwritten digits. *Pattern Recognition*, 45(4), 1318-1325. <https://doi.org/10.1016/j.patcog.2011.09.021>