Handwritten Bangla Digit Recognition using Sparse Representation Classifier

Haider Adnan Khan, Abdullah Al Helal and Khawza I Ahmed
Department of Electrical and Electronic Engineering
United International University, Dhaka 1209, Bangladesh
Email: {hkhan, helal, khawza}@eee.uiu.ac.bd

Abstract—**We present a framework for handwritten Bangla digit recognition using Sparse Representation Classifier. The classifier assumes that a test sample can be represented as a linear combination of the train samples from its native class. Hence, a test sample can be represented using a dictionary constructed from the train samples. The most sparse linear representation of the test sample in terms of this dictionary can be efficiently computed through ℓ_1-minimization, and can be exploited to classify the test sample. We applied Sparse Representation Classifier on the image zone density, an image domain statistical feature extracted from the character image, to classify the Bangla numerals. This is a novel approach for Bangla Optical Character Recognition, and demonstrates an excellent accuracy of 94% on the off-line handwritten Bangla numeral database CMATERdb 3.1.1. This result is promising, and should be investigated further.**

Index Terms—**Sparse Representation Classifier, Bangla Optical Character Recognition, Handwritten character recognition, Digit recognition**

I. Introduction

Handwritten digit recognition is a classical pattern recognition problem, and is widely used in different applications such as reading postal codes, bank checks, and numeral forms. Bangla is one of the most spoken languages in the world, and is spoken by more than 200 million native speakers. Consequently, automatic recognition of Bangla characters or numerals has a great importance. Different languages have different alphabets or scripts, and hence present different challenges for automatic character recognition. For instance, Bangla uses a Sanskrit based script, and is fundamentally different from English, a Latin based script. So, the accuracy of a character recognition algorithm may vary significantly depending on the script. Therefore, handwritten Bangla digit recognition algorithms should be investigated with due importance.

Liu and Suen [1] benchmarked the accuracy of recognition rate of handwritten Bangla digits on a standard data set, namely the ISI database of handwritten Bangla numerals [2], with 19392 training samples and 4000 test samples for 10 classes, i.e., 0 to 9. The reported accuracy is 99.4%. Such high accuracy is attributed to the extracted features based on gradient direction, and some advanced normalization techniques. Surinta et al. [3] proposed using a set of features such as the contour of the handwritten image computed using 8-directional codes, distance computed between hotspots and black pixels, and intensity of pixel space of small blocks. Each of these features is fed to a nonlinear SVM classifier separately, and the final decision is based on majority voting. The data set used in [3] is composed of 10920 examples, and the method achieves an accuracy of 86.8%. Xu et al. [4] used a hierarchical Bayesian network which takes the images directly as the network input, and classifies them using a bottom-up approach. An average recognition accuracy of 87.5% is achieved with a data set of 2000 handwritten sample images.

This paper proposes the use of Sparse Representation Classifier (SRC), and has shown to achieve an accuracy of 94% as the aforementioned classifier is applied on a data set of 6000 samples. SRC is first used in face recognition problem [5], and since then, has been applied in various classification problems such as vehicle tracking [6], speaker recognition [7], classification of hyperspectral imagery [8], and cancer detection [9]. To the best of our knowledge, SRC has never been applied to the problem of Bangla handwritten digit recognition. Although the result reported in this paper is early in the sense that it only

978-1-4799-5180-2/14/$31.00 ©2014 IEEE

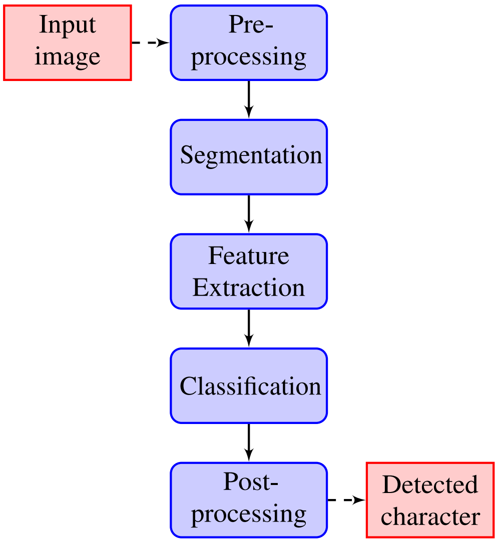


Fig. 1. Flowchart of a typical OCR system.

utilizes the pixel based zone density of the image as a feature, it shows promising sign underscoring the strength of SRC in dealing with Bangla Optical Character Recognition (OCR).

The rest of the paper is organized as follows. Section II summarizes the processes involved in the recognition task. Section III describes the zone density feature used for the recognition. Section IV introduces Sparse Representation Classifier (SRC). Tests and Results are discussed in Section V. Finally, concluding remarks are made in Section VI.

II. Methodology

The task of OCR is to detect and identify characters from the image of a text document, and mapping them into character codes that are computer readable and usable for further text processing.

A typical workflow to recognize characters from an image document is shown in Fig. 1. This includes the following steps:

1) Pre-processing: The input image is passed through a number of preliminary processing or pre-processing steps. The objective of pre-processing is to facilitate the OCR system to operate with higher accuracy. This can be achieved through a series of operations.
   a) Binarization: In order to convert the gray scale image into a binary image, the document image is thresholded. Image thresholding can be global or local (adaptive). Global image thresholding uses only one threshold value for the entire image while the local (adaptive) thresholding chooses different threshold values, according to the local information, for different image segments.
   b) Noise reduction: Noise reduction improves the quality of the image. Usually two common approaches are taken for noise reduction: 1) image filtering such as Wiener filter, Gaussian filter, and median filter, and 2) morphological operations such as erosion and dilation.
   c) Normalization: Normalizing inter-user and intra-user variability due to character size or choice of font family such as boldface is always a good idea. Common normalization steps include stroke width normalization or thinning, and normalization of aspect ratio and size of the image.
   d) Skew correction: Skew correction methods are employed in order to align the image document. Major approaches for skew detection include correlation, projection profiles, and Hough transform.
   e) Slant removal: The slant of the handwritten texts depends on the user. Slant removal methods are used to reduce the variability due to different writing styles, and to normalize all the characters to a standard form.
2) Segmentation: The objective of image segmentation in OCR system is to extract isolated characters from the image document. The segmentation step involves the following operations: text line detection, word extraction, and character segmentation. Segmentation of the characters to be identified is commonly performed in a top-down fashion — line segmentation is performed first, followed by word segmentation, and then by character segmentation.
3) Feature Extraction: In feature extraction step, the segmented character is transformed into

a set of features called feature vector. Each character is represented by its feature vector. Feature extraction provides dimensionality reduction, extracts the relevant information from the character image, and facilitates better separation and identification of different characters in the feature space.

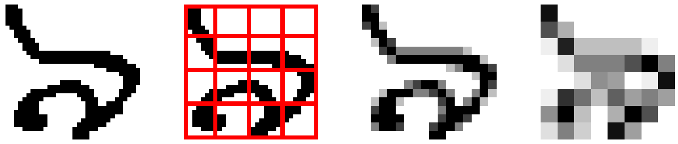
4) Classification: A classification scheme provides a decision rule to identify a character based on its features vector. This task can be achieved by exploiting machine learning approaches and standard classifiers such as Artificial Neural Network (ANN), K-Nearest Neighbor (KNN), Hidden Markov Model (HMM), and Support Vector Machine (SVM).
5) Post-processing: Dictionary-based approaches and context can be used to improve recognition rate, e.g., by correcting misspelling and choosing appropriate words.

In this work, we address the problem of identifying a set of segmented and binarized handwritten Bangla numerals. Therefore, we are concentrating only on Feature Extraction and Classification steps of the OCR system, which are discussed in greater details in Sections III and IV.

III. Feature Extraction

Handwritten characters usually demonstrate a high degree of variability due to different writing styles. Hence, selecting an appropriate set of features to represent and classify handwritten characters is not a trivial task. Feature extraction methods commonly exploit statistical features, structural features, global transformation, and moment based features for character recognition. In this paper, we used zone density, a statistical attribute of the character, for digit classification.

In order to extract zone density feature, we applied zoning on the input image. Zone based features have been proposed in the literature of OCR with successful outcome in different languages [10], [11]. The objective of zoning is to extract the local characteristics instead of the global characteristics. Zoning splits the image into $M \times N$ zones. Feature is extracted from each zone separately. For each zone, zone density is calculated as the normalized number of foreground pixels.



(a) (b) (c) (d)

Fig. 2. Illustration of zoning. (a) Original image. (b) 4×4 zoning. (c) 16×16 zone density. (d) 8×8 zone density. Images are scaled to 32×32 for illustrative purposes.

Zone density performs as a down sampled or low resolution version of the original character image. Fig. 2 illustrates the effect of zone density calculation for two different dimensions of zoning. This has the additional advantage of reducing feature dimension. For example, an image of 32×32 pixels contains 1024 features. Using 8×8 zoning, we get 64 zones, hence 64 different zone densities or features.

Additionally, zone density uses averaging (mean filtering) on all pixel values of each zone. This acts as a low pass filter, and reduces the effects of small differences in character image due to inter-user variability.

IV. Sparse Representation Classifier

We denote the image of each numeral as $\mathbf{P}(x,y)$, where the set of (x,y) consists of a grid of 2D pixels. Thus we want to identify an unknown $\mathbf{P}(x,y)$ as one of the $n_c = 10$ classes of Bangla numerals using available training samples where the true classes are known and labeled.

Given a set of n training images, we can extract features from any training sample $\mathbf{P}_j(x,y)$, and represent it by a *feature vector* $\mathbf{b}_j \in \mathbb{R}^m$ in a vector space. Thus each training sample is an *atom*, and the complete training set forms a *dictionary* $A \in \mathbb{R}^{m \times n}$:

$$\mathbf{A} = [\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n]$$

A simple assumption that images of the same class lie in a linear subspace has been utilized to obtain good accuracy in face recognition [12]. We model the feature vector $\mathbf{b}$ of a test image of class i as a linear combination of the feature vectors of the same class.

Since we will not know the native class of a test sample during training, we express a test sample of class i in terms of all training samples,

$$\mathbf{b} = a_1\mathbf{b}_1 + a_2\mathbf{b}_2 + \cdots + a_n\mathbf{b}_n = \mathbf{A}\mathbf{x}_r, \qquad (1)$$

$a_j \in \mathbb{R}$, $j = 1, 2, \ldots, n$. It readily follows from our model that only class i will contribute to non-zero coefficients a_j. Thus the solution vector

$$\mathbf{x}_r = [a_1, a_2, \ldots, a_n]^\mathsf{T} \tag{2}$$

contains the identity of the test sample.

The traditional approach of solving the linear system in (1) does not work, since the number of features m and the number of training samples n need not be equal. In case of $m > n$, the resulting over-determined system has generally no solution; in case of $m < n$, the resulting under-determined system has infinitely many solutions. Thus in either case, there will be no unique solution.

Observing the model (1) and the form of the solution (2) closely reveals that the solution is *sparse*, since a lot of the coefficients in $\mathbf{x}_r$ will be zero, making the model a *sparse representation* of the test sample in terms of the training samples. Intuition tells us to look for the most sparse solution with the most zeroed elements in the solution

$$\mathbf{x}_{\ell_0} = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_0, \quad \text{subject to} \quad \mathbf{A}\mathbf{x} = \mathbf{b}, \tag{3}$$

where $\|\cdot\|_{\ell_0}$ is the ℓ_0-norm indicating the number of non-zero coefficients in the operand vector. That solving (3) requires exhaustive search [13] has motivated the development of a theorem that states that a sparse enough vector $\mathbf{x}_r$ can be found by solving the following program [14]

$$\mathbf{x}_{\ell_1} = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_1, \quad \text{subject to} \quad \mathbf{A}\mathbf{x} = \mathbf{b}, \tag{4}$$

where $\|\cdot\|_{\ell_1}$ is the ℓ_1-norm indicating the sum of non-zero coefficients of the operand vector.

In an ideal scenario, only coefficients from the native class of a test sample will contribute to the sparse representation. Hence the unique membership of the non-zero coefficients is the identity of the test sample.

In real cases, however, coefficients from other classes contribute to the sparse representation, making the detection not straight-forward. As an example, Fig. 3 demonstrates that the sparse solution for a given character image is concentrated on its true class, marked by green, but other classes show non-zero coefficients as well, marked by red.

Coefficients from a single class provide information about how the class contributes to the solution.

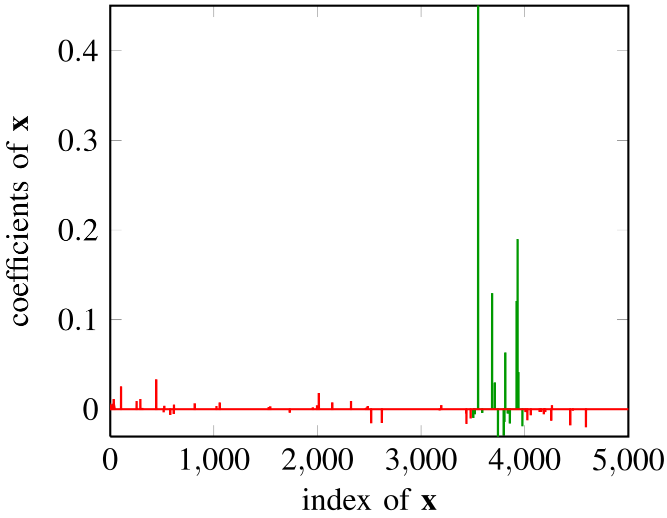


Fig. 3. Sample sparse solution. Green indicates true class, red false classes.

This motivates us to compute ℓ_1-norm of coefficients from a single class: $\delta_i(\mathbf{x}_{\ell_1})$, where $\delta_i(\mathbf{x}) \in \mathbb{R}^n$ is a vector generated from $\mathbf{x}$ whose only non-zero elements are those of $\mathbf{x}$ associated with class i. Consequently, the identity of the test sample can be found by maximizing (normalized) ℓ_1-norm of $\delta_i(\mathbf{x}_{\ell_1})$:

$$\text{detection}(\mathbf{b}) = \arg\max_i \frac{\|\delta_i(\mathbf{x}_{\ell_1})\|_1}{\|\mathbf{x}_{\ell_1}\|_1} \tag{5}$$

For example, $\delta_i(\mathbf{x}_{\ell_1})$ for the character image under consideration in Fig. 3 is shown in Fig. 4.

V. Experiments and Results

A. Data Set

We evaluated our method on the Handwritten Bangla Numeral Database CMATERdb 3.1.1 [15] collected from CMATERdb[1]. This data set contains 6000 images of unconstrained handwritten isolated Bangla numerals. Each digit has 600 binary images of 32×32 pixels.

Some sample images of the database are illustrated in Fig. 5. Visual inspection depicts that there is no visible noise. However, variability in writing style due to user dependency is quite high. The data set was split into a train set and a test set for the

[1]Center for Microprocessor Applications for Training Education and Research (CMATER) research laboratory, Jadavpur University, Kolkata, India. https://code.google.com/p/cmaterdb

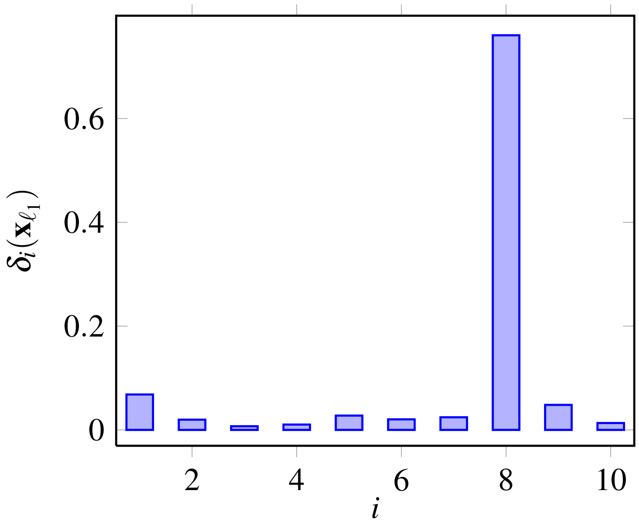


Fig. 4. Contribution of each class $\delta_i(\mathbf{x}_{\ell_1})$ in sparse solution. Maximum indicates detected class.

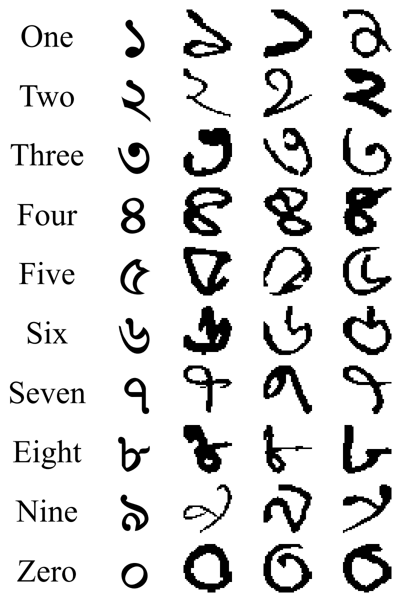


Fig. 5. Sample images: column 1 indicates the actual digit, column 2 demonstrates the numeral in 'SutonnyMJ' Bangla font, and columns 3–5 illustrate randomly selected images of the digit.

evaluation of the method. The train set consists of 5000 images (500 randomly selected images of each digit). The test set consists of the remaining 1000 images.

B. Feature Dimension

We used zone density feature for handwritten numeral recognition. Choosing the appropriate zone dimension (values of M and N) can be critical as demonstrated in Section V-C. If we select too many zones, we get very little (for example, in case of 16×16 zones in this data set), or no (in case of 32×32 zones) feature reduction and image smoothing.

On the other hand, using a zoning dimension too small such as 1×1 or 2×2 will ignore almost all local variation leaving only global information. For our experiment, we used 32×32 (the original image), 16×16, 8×8 and 4×4 zones, and compared the detection performance.

TABLE I
DETECTION ACCURACY FOR DIFFERENT FEATURE DIMENSIONS.

Zone Dimension	Feature Dimension	Accuracy
32×32	1024	78.5%
16×16	256	86.5%
8×8	64	94.0%
4×4	16	89.2%

C. Results

Table I demonstrates the detection accuracy of digit recognition for different feature sets according to the zone dimensions. For the original 32×32 image (with no zoning), the detection accuracy was quite low — 78.5%. However, the detection accuracy gets significantly better as we apply zoning on the character image — 16×16 zoning yields 86.5% detection accuracy, and 8×8 zoning yields the best accuracy of 94.0%. When zone dimension is reduced further, the detection accuracy gets lower, e.g., 4×4 zoning shows 89.2% accuracy. This result reflects the fact that while zoning helps in feature dimension reduction, too much feature reduction can lead to a decline in the detection accuracy as explained in Section III.

For the best performing feature set (8×8 zones), we additionally calculated the detection accuracy for each digit separately. The result is displayed in Table II, which demonstrates that digits such as four and eight show very high degree of accuracy while other ones such as three and five show relatively low accuracy of detection.

VI. CONCLUSION

We used a novel classifier, Sparse Representation Classifier (SRC), for off-line isolated handwritten

TABLE II
DETECTION ACCURACY FOR DIFFERENT CLASSES.

Digit	Accuracy
1	94
2	93
3	89
4	99
5	87
6	92
7	98
8	99
9	92
0	97

Bangla numeral recognition. The method demonstrates an excellent result with 94% overall accuracy. This result is very promising, and is likely to improve if pre-processing techniques such as normalization, skew correction, and slant removal are applied. Further improvement may be achieved with the use of appropriate features specific to the Bangla digits, and different variants of SRC such as regularized SRC and kernel SRC. Comparison with other conventional classifiers should be considered in future as a continuation of this work. The results should also be verified for other standard handwritten character databases such as the ISI database of handwritten Bangla numerals.

REFERENCES

[1] C.-L. Liu and C. Y. Suen, “A new benchmark on the recognition of handwritten bangla and farsi numeral characters,” *Pattern Recognition*, vol. 42, no. 12, pp. 3287–3295, 2009, new Frontiers in Handwriting Recognition.

[2] B. B. Chaudhuri, “A complete handwritten numeral database of bangla – a major indic script,” in *Tenth International Workshop on Frontiers in Handwriting Recognition*, Oct. 2006.

[3] O. Surinta, L. Schomaker, and M. Wiering, “A comparison of feature and pixel-based methods for recognizing handwritten bangla digits,” in *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*. IEEE, Aug. 2013, pp. 165–169.

[4] J.-w. Xu, J. Xu, and Y. Lu, “Handwritten Bangla digit recognition using hierarchical Bayesian network,” in *Intelligent System and Knowledge Engineering, 2008. ISKE 2008. 3rd International Conference on*, vol. 1. IEEE, Nov. 2008, pp. 1096–1099.

[5] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, “Robust face recognition via sparse representation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[6] X. Mei and H. Ling, “Robust visual tracking and vehicle classification via sparse representation,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 11, pp. 2259–2272, 2011.

[7] J. M. K. Kua, E. Ambikairajah, J. Epps, and R. Togneri, “Speaker verification using sparse representation classification,” in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 4548–4551.

[8] Y. Chen, N. M. Nasrabadi, and T. D. Tran, “Hyperspectral image classification using dictionary-based sparse representation,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 10 PART 2, pp. 3973–3985, 2011.

[9] A. A. Helal, K. I. Ahmed, M. S. Rahman, and S. K. Alam, “Breast cancer classification from ultrasonic images based on sparse representation by exploiting redundancy,” in *16th International Conference on Computer and Information Technology, ICCIT 2013*, Mar. 2014.

[10] C.-L. Liu, K. Nakashima, H. Sako, and H. Fujisawa, “Handwritten digit recognition: investigation of normalization and feature extraction techniques,” *Pattern Recognition*, vol. 37, no. 2, pp. 265–279, 2004.

[11] B. V. Dhandra, G. Mukarambi, and M. Hangarge, “Zone based features for handwritten and printed mixed Kannada digits recognition,” *IJCA Proceedings on International Conference on VLSI, Communications and Instrumentation (ICVCI)*, no. 7, pp. 5–8, 2011.

[12] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, “Eigenfaces vs. fisherfaces: Recognition using class specific linear projection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, 1997.

[13] E. Amaldi and V. Kann, “On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems,” *Theoretical Computer Science*, vol. 209, no. 1–2, pp. 237–260, 1998.

[14] D. L. Donoho, “For most large underdetermined systems of linear equations the minimal l1-norm solution is also the sparsest solution,” *Communications on Pure and Applied Mathematics*, vol. 59, no. 6, pp. 797–829, 2006.

[15] N. Das, R. Sarkar, S. Basu, M. Kundu, M. Nasipuri, and D. K. Basu, “A genetic algorithm based region sampling for selection of local features in handwritten digit recognition application,” *Applied Soft Computing*, vol. 12, no. 5, pp. 1592–1606, 2012.