A Comparison of Feature and Pixel-based Methods for Recognizing Handwritten Bangla Digits

Olarik Surinta, Lambert Schomaker and Marco Wiering
Institute of Artificial Intelligence and Cognitive Engineering, University of Groningen
Nijenborgh 9, Groningen, The Netherlands
Email: {o.surinta, l.r.b.schomaker, m.a.wiering}@rug.nl

Abstract—We propose a novel handwritten character recognition method for isolated handwritten Bangla digits. A feature is introduced for such patterns, the contour angular technique. It is compared to other methods, such as the hotspot feature, the gray-level normalized character image and a basic low-resolution pixel-based method. One of the goals of this study is to explore performance differences between dedicated feature methods and the pixel-based methods. The four methods are compared with support vector machine (SVM) classifiers on the collection of handwritten Bangla digit images. The results show that the fast contour angular technique outperforms the other techniques when not very many training examples are used. The fast contour angular technique captures aspects of curvature of the handwritten image and results in much faster character classification than the gray pixel-based method. Still, this feature obtains a similar recognition compared to the gray pixel-based method when a large training set is used. In order to investigate further whether the different feature methods represent complementary aspects of shape, the effect of majority voting is explored. The results indicate that the majority voting method achieves the best recognition performance on this dataset.

Keywords-Handwritten Bangla digit recognition, Character recognition, Feature extraction technique, Pixel-based method, Classification, Support vector machines

I. INTRODUCTION

The main challenge in handwritten character classification is to deal with the enormous variety of handwriting styles as a result of different writers. Furthermore, some complex handwriting scripts comprise different styles for writing words. In some of them characters are written isolated from each other, (*e.g.*, Thai, Laos and Japanese), in some they are cursive, and in some the characters are connected (*e.g.*, English, Indian and Arabic). This challenge is recognized by many researchers [1]–[3]. The large variety of writing styles, writing persons, and the complicated features of the handwritten characters are very challenging for accurately classifying the characters.

A large number of studies investigated the problem of handwriting recognition based on the MNIST dataset [4]. The MNIST dataset was modified from the original NIST database [4]. This dataset is nowadays used as a standard benchmark for testing machine learning techniques and pattern recognition methods [1], [2]. There are $60,000$ handwritten digit images for training and $10,000$ test images. The size of these handwritten digit images is normalized and the digits are centered in a fixed-size image to fit into a 28×28 pixel space [2], [4]. Furthermore, the handwritten images are

completely separated from the background. Although this is a large dataset, it is very clear and researched extensively.

This paper focuses on the recognition of handwritten *Bangla* (or Bengali) digits, which is the second most popular language in India and Bangladesh [5]. The dataset contains different kinds of background and a variety of pixel space resolutions. Various example digits of handwritten Bangla are shown in Fig. 1. It is clear, that when compared to the MNIST dataset, Bangla digits are more complicated and there is more style diversity [6]. For instance, the curly tails in Bangla characters makes the definition of a stable bounding box problematic.

A feature extraction technique can play an important factor for getting high accuracies in handwritten character recognition systems [7]. Different feature extraction techniques have been used to train a recognizer for the handwritten Bangla digit dataset. However, also pixel-based methods that directly use the pixels have been explored. In one approach, a pixel-based method, pixels are represented as data in high dimensional input space. Wen *et al.* [8] and Roy *et al.* [9] presented the original handwritten Bangla digit without computing any features and normalized the original image into a 16×16 and 28×28 pixel space, respectively. The number of handwritten digits used in their experiments were $16,000$ and $10,677$ records, respectively. Wen *et al.* [8] used the support vector machine (SVM) classifier and obtained a recognition rate of 86.1%. Roy *et al.* [9] used the multilayer perceptron (MLP) for the recognition and obtained a recognition rate of 92.1%

In the feature-based approach, an automatic feature extraction technique extracts unique information from the handwrit-
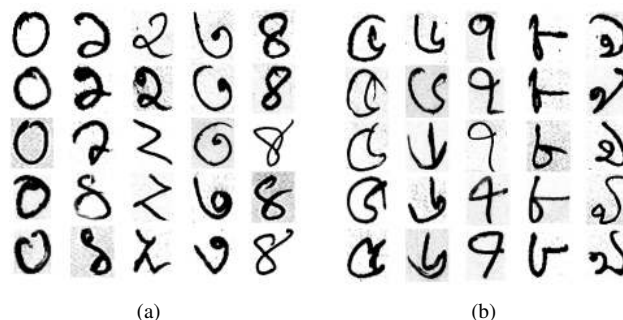

Fig. 1. A variety of handwritten Bangla digit samples. (a) Set of numbers from 0 to 4 and (b) from 5 to 9. Note the large differences between the examples on the first and last rows.

ten image. The number of resulting features from the feature extraction technique is often smaller than in the pixel-based method. Surinta *et al.* [7] proposed the hotspot technique for the handwritten Bangla digit dataset, from which $8,635$ Bangla digits were used in the training set. They used 40×40 pixels of binary images. The distance between the black pixels and the hotspot in each direction gives the feature values. The hotspot technique provided feature vectors that were classified using k-nearest neighbors (kNN) and obtained a recognition rate of 90.1%. Wen *et al.* [8] proposed the KPS technique. Their technique combined a Kirsch mask and principal component analysis (PCA). The dimensionality reduction was used, because the Kirsch mask extracted $1,280$ dimensional inputs. After PCA they used the SVM classifier and obtained an accuracy of 95.1%. Basu *et al.* [10] presented the shadow, the centroid and the longest-run feature extraction technique. It provided 76 dimensional inputs and $6,000$ records were used, which were scaled to 32×32 pixel space. They used the MLP classifier and obtained an accuracy of 96.7%.

Contributions. We present a novel method that obtains state-of-the-art performance on the isolated handwritten Bangla digit dataset. We have compared four different techniques. The first feature is the contour angular technique that computes the contour of the handwritten image using 8-directional codes, while counting the co-occurrences of angles along the ink trace. The second feature is the set of distance values that is computed between the hotspots and the black pixels of the handwritten image [7]. The third feature uses intensities of the pixel space from small blocks. The last feature is a gray pixel-based method that uses the whole handwritten image as the input [5]. Finally, we use a majority voting technique to combine the outputs of the four different classifiers and to obtain the highest recognition accuracy.

Outline of this paper. This paper has been organized in the following way. *Section II* describes the handwritten Bangla digit dataset. This dataset was preprocessed using a binarization, normalization and thinning algorithm. *Section III* describes two major types of feature computation, namely different feature extraction techniques and two pixel-based methods. The multi-class SVM used for classification is described in *Section IV*. *Section V* shows the experimental results. The last section discusses the significant findings from this study and suggests directions for future work.

II. Dataset Preparation

We evaluate our recognition methods on the handwritten Bangla digit dataset[1]. Our dataset is composed of $10,920$ examples of the numbers 0 to 9 (10 classes). The most distinctive aspect of this dataset is the large variety of handwriting styles (Fig.1). Because of the different handwriting styles, two of the numbers look similar as shown in Fig. 2.

In order to prepare the handwritten images for our feature extraction methods, our system executes a number of preprocessing steps. First, the data in the handwritten Bangla digit
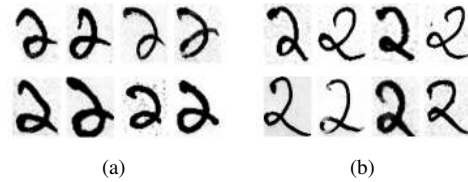
Fig. 2. Similarities between different handwritten Bangla digits. (a) The images of number 1, and (b) number 2.

dataset contain different kinds of backgrounds, some of which are clear but most are not clear and even quite noisy. The four background removal algorithms investigated in this study were those of Otsu, Niblack, Sauvola and Wolf [11], [12]. The results of handwritten images after applying the binarization methods are shown in Fig. 3. In this experiment, Otsu's algorithm (Fig. 3(b)) was the best algorithm to transform a gray image into a binary image without noise.
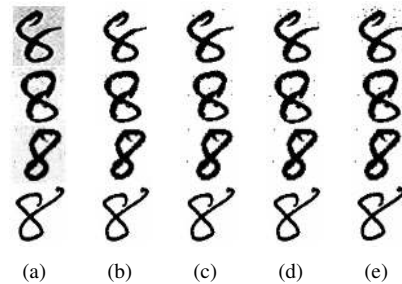

Fig. 3. Results of handwritten images after applying the binarization methods. (a) The handwritten image before removing background noise, (b) background noise is removed by Otsu's algorithm, (c) Wolf's algorithm, (d) Sauvola's algorithm and (e) Niblack's algorithm.

The second problem with the handwritten Bangla digit dataset is that numbers were scanned into digital images at different resolutions. The Bicubic interpolation, which is an efficient normalization algorithm [13], [14], was used to normalize the handwritten image to fit into a 28×28 pixel space, which yields images with quality from good to outstanding.

Finally, the last process in preparation of the dataset is thinning. This technique is used in order to create images which are uniformly thin, as the current dataset (Fig.1) displays a large variety of ink thickness. The results from the thinning technique are shown in Fig. 4. Other researchers such as Pal *et al.* [15] and Liu and Suen [16] have not applied thinning, as this procedure is not useful for their feature extraction techniques. We will use thinning for our two feature extraction methods, but not for the pixel-based methods.
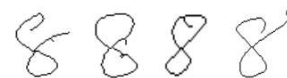

Fig. 4. The handwritten images after applying the thinning technique.

III. CREATION OF A FEATURE VECTOR

In order to create a feature vector, we study two kinds of techniques. In both, the relevant information is extracted from the handwritten character image and transformed into vector data [17]. The two major techniques include feature extraction techniques and pixel-based methods. Our feature extraction techniques reduce the size of the feature vector. On the other hand, the pixel-based methods just use individual pixel values as a feature vector. These techniques are described below.

A. Feature Extraction Techniques

1) The Contour Angular Technique (CAT): What is needed is a feature that can be computed conveniently and which captures more shape details than the angle along the contour of the ink trace. A proposed solution is to capture aspects of curvature. The approach is related to the usage of angular co-occurrence in the Hinge feature [18] in writer identification.

The CAT implementation is, unlike the Hinge, directed at the classification of characters and is a fast implementation of quantized angle co-occurrence computation. The technique consists of two stages. In the first stage, the method divides the handwritten character into 16 non-overlapping blocks and considers the contour of the handwritten image as 8-directional codes, see Fig. 6(a). This setting computes 128 features. *First*, the starting point S_i for each block is identified. It is searched along the edges of image blocks, first along the upper, the right, the bottom until finished at the left edge. The first found black pixel is the starting point S_i (Fig. 5(b)). *Second*, 8-directional codes are used for identifying the contour of the neighbor pixels. Sometimes there are multiple neighbor pixels, for this reason a queue is used to arrange the neighbor pixels, which are all used to update the directional-code histogram.
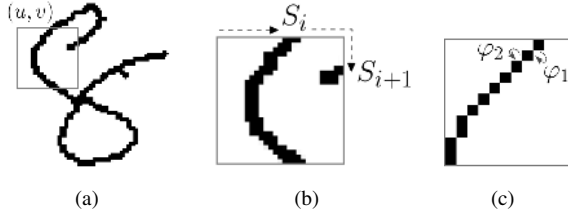

Fig. 5.   The starting point S_i of the handwritten image block. (a) The handwritten image block (u, v) from the whole handwritten image. (b) The starting point S_i of the handwritten image is searched along the (u, v) borders. (c) The angular co-occurrence for which angles φ_1 and φ_2 are computed.

In the second stage, the contour of the handwritten image is computed. The method considers the angles φ_1 and φ_2 (Fig. 5(c)) that move from S_i until the last pixel in each block and then counts the co-occurring angles in a two-dimensional array indexed by φ_1 and φ_2. The result is the angular co-occurrence histogram with 64 elements. The discrete angle co-occurrence histogram approximates the angular co-occurrence probability along the contours. By combining the outputs of both stages, the CAT feature extraction method creates feature vectors of size 192.

2) The Hotspot Technique (HOT): The distance between evenly spaced hotspots and the closest black pixels in each direction is used to describe the whole handwritten image. For each direction d_i, the distance D_{si} between the hotspot and the closest black pixel (x_i, y_i) of the handwritten image is found. The distance is set to d_{max} when black pixels do not exist in that direction (Fig. 6(b)). Here, $d_{max} = 20$ is used. The distance value is computed by (Eq. (1)):

$$
D_{si} = \begin{cases} \sqrt{(x_s - x_i)^2 + (y_s - y_i)^2} & if \ (x_i, y_i) \ exists, \\ d_{max} & else \end{cases}
$$

(1)

Where (x_s, y_s) is the coordinate of the hotspot and (x_i, y_i) is the coordinate of the closest black pixel of the handwritten image in the specified direction d_i.
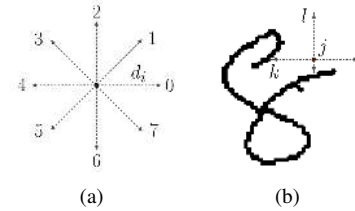

Fig. 6.   The hotspot technique for the handwritten images. (a) The 8-directional codes for representing the directions. (b) The feature values of the hotspot. j is the coordinate of the hotspot (x_s, y_s), k is a distance value D_{si} when (x_i, y_i) exists and l is a distance value d_{max} when D_{si} does not exist.

The parameters of the hotspot technique [7] include the number of hotspots and the number of directional codes. The classification rate of 90.1% was obtained from handwritten Bangla digits with the best setting using 100 features. This extractor is designed with 25 evenly spaced hotspots. The directions of the hotspots are defined by the 4-directional codes $d_i \in \{0, 2, 4, 6\}$.

B. Pixel-Based Methods

1) Gray Pixel-Based Method (GPB): The Gray (28×28) pixel-based method uses the raw pixel intensities of the handwritten images to preserve the handwritten image without destructing subtle intensity gradients on the edges of the inktrace [9]. The size of the handwritten image is resized to the corresponding resolution, 784 feature values are computed.

2) Black and White Down Scaled (9×9) Method (BWS): This simple feature technique is useful to compute a base performance. The black and white handwritten image is partitioned into 9×9 non-overlapping blocks. From each block the number of black pixels is computed, resulting in 81 features.

IV. SUPPORT VECTOR CLASSIFICATION

The SVM algorithm is very useful for two-class classification problems [19]. The SVM finds the optimal hyperplane that is the best separation of input vectors belonging to different classes. The optimal hyperplane should be as far away from the closest data points of both classes as possible.

The training set is $(\mathbf{x}_i, y_i)$, $i = 1, ..., l$, where $\mathbf{x}_i \in \mathbf{R}^n$ with corresponding labels $y_i \in \{1, -1\}$. It can be split by the hyperplane $\mathbf{w}^T\mathbf{x} + b = 0$, where $\mathbf{w}$ is the weight vector and b is the bias. The optimal separating hyperplane obtains the biggest distance to the closest positives $\mathbf{w}^T\mathbf{x} + b = +1$ and negatives $\mathbf{w}^T\mathbf{x} + b = -1$. The linear classifier is uncomplicated, but it is unsuitable for the variety of input vectors and classes in our dataset.

A. Non-linear SVMs for Multi-Class Problems

The SVM can be extended to deal with more than 2 classes by constructing and combining several binary classifiers [20]. We use the one-vs-all strategy. The problem consists of choosing the kernel function and tuning a variety of parameters [21]. We chose the radial basis function (RBF) kernel as a non-linear similarity function, because it usually outperforms other kernels [22]. The RBF kernel is given by:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2\right)$$

The hyperparameters that need to be tuned in the SVM with the RBF kernel include the cost parameter (C) and the gamma parameter (γ). The C parameter has a significant effect on the decision boundary. It controls the width of the margin. The γ parameter directly affects overfitting. This causes large γ values to increase the number of support vectors [21].

B. Model Selection

We used regular grid search for exploring the two dimensional parameter space. The logarithmic scale is chosen: for C values in $\{2^{-3}, 2^{-2}, \ldots, 2^4\}$ are tried and for γ we tried values in $\{2^{-6}, 2^{-5}, \ldots, 2^5\}$. The goal is to estimate the accuracy of the classifier for each point on the grid. In order to prevent over-fitting, cross validation is used. The training set is divided into k-folds, one fold is used as test set, and $k - 1$ folds are used as training set. This is then repeated k times.

C. Normalization

It is very important to normalize the feature vector before applying SVMs. This is to avoid attributes of the feature vector in large numeric ranges. Therefore, the entire training and test dataset are scaled with the same normalization method. In our experiments, the features are scaled to the range $[0, 1]$.

V. Experimental Results

We have used 10-fold cross validation to evaluate the results of the handwritten Bangla digit recognition methods. The best values of the C and the γ parameters, which were found by grid search, are chosen and used to train a model which is evaluated on the test set. The result of this process is the mean accuracy and the standard deviation (σ).

We used training set sizes of $10, 20, 30, 40$ and 90%, respectively of 10920 examples in total. The summary of results is given in Table I. The recognition rate of GPB was quite low at 10% of the training set size for which it obtained 90.5% accuracy. The CAT feature obtained 92.2% accuracy when the dataset was decreased to 10%. The recognition rate of the CAT

TABLE I
THE RECOGNITION RATES OF THE DIFFERENT METHODS

Training data	Recognition rate (%)			
	CAT	HOT	GPB	BWS
10%	**92.2** $(\sigma = 0.9)$	88.0 $(\sigma = 1.7)$	90.5 $(\sigma = 1.3)$	88.1 $(\sigma = 2.1)$
20%	**94.3** (1.2)	90.3 (1.4)	93.6 (1.5)	91.0 (2.2)
30%	**95.4** (0.9)	91.8 (1.4)	94.8 (1.2)	93.5 (1.5)
40%	**95.9** (1.0)	92.7 (1.2)	**95.9** (1.3)	94.1 (0.9)
90%	96.0 (0.8)	92.7 (0.8)	**96.4** (0.9)	94.7 (1.1)

feature is significantly higher than the GPB pixel-based method for this small amount of training data. When all training examples are used, the best pixel-based method is slightly more powerful than the best feature extraction technique, although this difference is not statistically significant. The GPB obtains a high accuracy of 96.4%. The results of the feature and the pixel-based methods for recognizing the Bangla digits are also shown in Fig. 7.

Furthermore, we used the unweighted majority vote method (UMV) [23] to combine the outputs from the four different SVM classifiers. The number of votes for each class is counted, and the class with the majority is selected as the output of the ensemble. A random method is used to choose between classes when they obtained the same number of votes. The accuracy increased to 96.8% when UMV is used. The results of UMV are shown in Table II and in Fig. 7. In both result tables, the difference between the best method and the second best method is only statistically significant with 10% of the training examples.

TABLE II
THE RECOGNITION RATES OF THE UNWEIGHTED MAJORITY VOTE FOR COMBINING SVM CLASSIFIERS

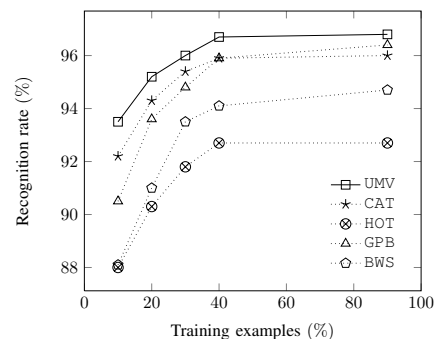| Training data	Recognition rate (%)	
	Best classifier	UMV
10%	92.2 $(\sigma = 0.9)$	**93.5** $(\sigma = 1.0)$
20%	94.3 (1.2)	**95.2** (1.0)
30%	95.4 (0.9)	**96.0** (0.8)
40%	95.9 (1.0)	**96.7** (0.9)
90%	96.4 (0.9)	**96.8** (0.6)

Fig. 7. The recognition rates of feature techniques and the majority vote for combining SVM classifiers on the handwritten Bangla digit dataset.

We were also interested in the computation time of training the SVM classifier. In this experiment, the used desktop computer is: Intel(R) Pentium(R) D CPU 3.00GHz. The computation times of the different training schemes are shown in Fig.

8. `CAT` is 13.5 times faster than `GPB`. We also measured the operational time it costs on average to perform the necessary operations for each method (e.g., converting to 28×28, binarization, thinning, feature extraction, and classification with the SVM). The `BWS` method is the fastest one with 29 ms used to classify a handwritten character image. The `CAT` and `HOT` methods require 37 ms and 35 ms, respectively. Finally, the slowest method is the `GPB` technique that consumes 73 ms to classify an image.
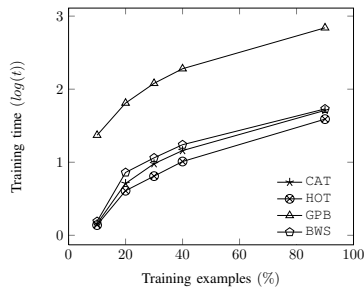

Fig. 8. Results of the time ($log(t)$) needed for training the SVM classifier with the different methods for computing features, where t is in seconds.

VI. CONCLUSION

In this paper we studied if feature extraction techniques can outperform pixel-based methods for handwritten character recognition. We described some advanced feature extraction techniques and evaluated the performance on the handwritten Bangla digit dataset. The techniques that were used include `CAT`, `HOT`, `GPB` and `BWS`, and the system used a support vector machine as a classifier to yield high accuracies. The best feature extraction technique `CAT` outperforms the best pixel-based method when the training dataset is not very large. When the training dataset size increases, the best pixel-based method slightly outperforms this feature extraction method. However, in terms of computation time the `CAT` feature extraction method outperforms the `GPB` pixel-based method, because the latter uses all pixels of the handwritten image. Finally, The majority voting technique increased the performance of the handwritten character recognition system. It obtained 96.8% accuracy with 90% of the training data.

In future work, we will collect a new handwritten Thai dataset including characters and digits. Then, we will use the best feature extraction techniques, and develop novel methods. Thus, we aim to provide a new benchmark for handwritten digit recognition, and obtain high accuracies on the handwritten Thai dataset, which is challenging because these characters also have curly extensions and shape variations, unlike the plain Arabic numerals in MNIST.

REFERENCES

[1] D. C. Ciresan, U. Meier, L. M. Gambardella, and J. Schmidhuber, "Deep big simple neural nets excel on handwritten digit recognition," *Neural Computation*, vol. 22, no. 12, pp. 3207–3220, dec. 2010.

[2] U. Meier, D. Ciresan, L. Gambardella, and J. Schmidhuber, "Better digit recognition with a committee of simple neural nets," in *Document Analysis and Recognition (ICDAR), 2011 International Conference on*, sept. 2011, pp. 1250–254.

[3] W. Song, S. Uchida, and M. Liwicki, "Comparative study of part-based handwritten character recognition methods," in *Document Analysis and Recognition (ICDAR), 2011 International Conference on*, sept. 2011, pp. 814–818.

[4] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Proceedings of the IEEE*, 1998, pp. 2278–2324.

[5] N. Das, B. Das, R. Sarkar, S. Basu, M. Kundu, and M. Nasipuri, "Handwritten Bangla basic and compound character recognition using MLP and SVM classifier," *Journal of Computing*, vol. 2, no. 2, 2010.

[6] T. Bhowmik, P. Ghanty, A. Roy, and S. Parui, "SVM-based hierarchical architectures for handwritten Bangla character recognition," *International Journal on Document Analysis and Recognition*, vol. 12, pp. 97–108, 2009.

[7] O. Surinta, L. Schomaker, and M. Wiering, "Handwritten character classification using the hotspot feature extraction technique," in *Pattern Recognition Applications and Methods, 2012. ICPRAM '12. First International Conference on*, feb. 2012, pp. 261–264.

[8] Y. Wen, Y. Lu, and P. Shi, "Handwritten Bangla numeral recognition system and its application to postal automation," *Pattern Recognition*, vol. 40, no. 1, pp. 99–107, 2007.

[9] K. Roy, S. Vajda, U. Pal, and B. Chaudhuri, "A system towards Indian postal automation," in *Frontiers in Handwriting Recognition, Ninth International Workshop on*, 2004, pp. 580–585.

[10] S. Basu, N. Das, R. Sarkar, M. Kundu, M. Nasipuri, and D. K. Basu, "An MLP based approach for recognition of handwritten 'Bangla' numerals," in *Computer Processing of Bangla, 2005. National Conference on*, feb. 2005, pp. 285–291.

[11] H. M. Som, J. M. Zain, and A. J. Ghazali, "Application of threshold techniques for readability improvement of Jawi historical manuscript images," *Advanced Computing: An International Journal*, vol. 2, no. 2, pp. 60–69, mar. 2011.

[12] C. Wolf, J.-M. Jolion, and F. Chassaing, "Text localization, enhancement and binarization in multimedia documents," in *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, vol. 2, 2002, pp. 1037–1040.

[13] J. Mahmud, "An intelligent feature analyzer for handwritten character recognition," in *Computational Intelligence for Modelling, Control and Automation, 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce*, vol. 2, 2005, pp. 763–769.

[14] M. Hamidi and A. Borji, "Invariance analysis of modified C2 features: case study-handwritten digit recognition," *Machine Vision and Applications*, vol. 21, pp. 969–979, 2010.

[15] U. Pal, N. Sharma, T. Wakabayashi, and F. Kimura, "Handwritten character recognition of popular south Indian scripts," in *Arabic and Chinese Handwriting Recognition*, ser. Lecture Notes in Computer Science, D. Doermann and S. Jaeger, Eds. Springer Berlin / Heidelberg, 2008, vol. 4768, pp. 251–264.

[16] 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.

[17] B. 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.

[18] M. Bulacu and L. Schomaker, "Text-independent writer identification and verification using textural and allographic features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 4, pp. 701–717, 2007.

[19] V. Vapnik, *Statistical learning theory*. Wiley, 1998.

[20] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Transactions on Neural Networks*, vol. 13, no. 2, pp. 415–425, 2002.

[21] A. Rossi and A. de Carvalho, "Bio-inspired optimization techniques for SVM parameter tuning," in *Neural Networks, 2008. SBRN '08. 10th Brazilian Symposium on*, oct. 2008, pp. 57–62.

[22] A. Abdullah, R. Veltkamp, and M. Wiering, "Fixed partitioning and salient points with MPEG-7 cluster correlograms for image categorization," *Pattern Recognition*, vol. 43(3), pp. 650–662, 2010.

[23] F. Moreno-Seco, J. Iesta, P. de Len, and L. Mic, "Comparison of classifier fusion methods for classification in pattern recognition tasks," in *Structural, Syntactic, and Statistical Pattern Recognition*, ser. Lecture Notes in Computer Science, D.-Y. Yeung, J. Kwok, A. Fred, F. Roli, and D. de Ridder, Eds. Springer Berlin / Heidelberg, 2006, vol. 4109, pp. 705–713.