Received: 9 May 2023 | Revised: 6 October 2023 | Accepted: 22 November 2023

DOI: 10.4218/etrij.2023-0188

ORIGINAL ARTICLE

ETRI Journal WILEY

Writer verification using feature selection based on genetic algorithm: A case study on handwritten Bangla dataset

Jaya Paul[1] | Kalpita Dutta[1] | Anasua Sarkar[1] | Kaushik Roy[2] | Nibaran Das[1]

[1]Department of Computer Science & Engineering, Jadavpur University, Kolkata, India

[2]Department of Computer Science, West Bengal State University, Barasat, India

Correspondence
Jaya Paul, Department of Computer Science & Engineering, Jadavpur University, India.
Email: jayapl2005@gmail.com

Abstract

Author verification is challenging because of the diversity in writing styles. We propose an enhanced handwriting verification method that combines handcrafted and automatically extracted features. The method uses a genetic algorithm to reduce the dimensionality of the feature set. We consider offline Bangla handwriting content and evaluate the proposed method using handcrafted features with a simple logistic regression, radial basis function network, and sequential minimal optimization as well as automatically extracted features using a convolutional neural network. The handcrafted features outperform the automatically extracted ones, achieving an average verification accuracy of 94.54% for 100 writers. The handcrafted features include Radon transform, histogram of oriented gradients, local phase quantization, and local binary patterns from interwriter and intrawriter content. The genetic algorithm reduces the feature dimensionality and selects salient features using a support vector machine. The top five experimental results are obtained from the optimal feature set selected using a consensus strategy. Comparisons with other methods and features confirm the satisfactory results.

KEYWORDS

genetic algorithm, handwriting analysis, n-quality consensus, text-dependent analysis, writer verification

1 | INTRODUCTION

Writer identification [1] and verification [2] from handwritten documents are the most important applications of forensic handwriting analysis in the criminal justice system [3]. Writer identification determines the author of a handwritten text from various possible writers based on handwriting. Writer verification consists of claiming the identity of the writer of a document. Writer identification is used in forensic science [4], biometric systems [5], historical document analysis, and security systems. Writer verification is mainly applied in forensic laboratories and relies on authentication [6]. It has been less explored than writer identification. Writer identification and verification by a human expert based on a handwritten document are extremely time-consuming and error-prone. Therefore, several automated methods have been developed for these tasks. Both online and offline methods are available. Online methods involve collecting written content in real time on specialized devices to calculate the character shapes and pen pressure. For offline methods, writing content is collected from scanned documents.

In this study, we focused on offline text-dependent Bengali writer verification, which presents several challenges, particularly owing to the lack of a standard Bangla dataset. The handwriting patterns of writers may vary according to the style, establishing interwriter variability. Additionally, handwriting samples from the same writer may differ extensively owing to various circumstances, such as mood, time, geographical location, or writing media, establishing intrawriter variability. These two types of variabilities are also known as in-between writer and between-writer, respectively [7].

The contributions of this study are as follows:

- A new dataset was constructed using 1180 images from 100 native Bengali writers and acquired at the document level. Experiments can be carried out on both the page and block levels.
- A genetic algorithm (GA) is used for the selection of representative features from extracted handcrafted features (histogram of oriented gradients—HOG, local binary pattern—LBP, local phase quantization—LPQ, and Radon transform). The performance of the models is tested for different classifiers.
- To overcome the GA probabilistic nature, a five-quality consensus strategy provides the best results when using the sequential minimal optimization (SMO) classifier.
- The performance of the proposed method using the handcrafted feature set is confirmed in comparison with AlexNet, a deep learning model that involves automatic feature extraction.

Figure 2 shows samples from writers 1 and 2, who wrote the same text but with different handwriting style depending on various essential factors, such as time, space, mood, and writing speed.

We developed an offline writer verification method that relies on different handwritten patterns. The proposed method was tested on our newly created handwriting dataset. The features considered in our analysis were based on the representations of writing orientation and textures: Radon transform, LBP, HOG, and LPQ. The proposed method combines features based on their performance on page fragments to improve the verification performance. We evaluated the feature performance using both the original features and a reduced representation regarding feature dimensionality after processing with a GA. The combination of features obtained from the GA for optimal feature selection improved the verification performance. Our method is inexpensive because it does not require special equipment for data acquisition. In addition, the corresponding model is straightforward to apply and achieves efficient verification of individual writers.

Because of the lack of handwritten samples from writers, better results could not be achieved using AlexNet. Our experimental setup was related to one-to-one decision making, where experts made their own decisions by analyzing small fragments of a paragraph to check its authenticity.

The remainder of this paper is organized as follows. Section 2 briefly describes related work. In Section 3, we detail the proposed method and development of a Bengali writer verification dataset. In addition, it presents the extraction of representative features from the newly developed dataset and GA-based feature dimensionality reduction applied to verification. Section 4 presents the evaluation results of the proposed and similar methods. Finally, we draw conclusions in Section 5.

2 | RELATED WORK

In this section, we discuss related work on writer verification. Handwriting has become a common biometric pattern in recent years [7]. Nevertheless, the digital world is open for research, and a digital future seems plausible.

Halder and others [8] combined various texture features to authenticate writers of isolated Bangla characters. They used 35,500 isolated Bangla characters from 500 documents for evaluation. Khan and others [9] introduced an offline text-independent writer verification method that processed partially damaged documents in handwritten Arabic text. Their approach relied on analyzing the shapes of individual characters, and they constructed a dataset called AHAWP, which included samples from 82 distinct writers. A convolutional neural network was developed to verify writers based on individual characters, achieving a notable writer verification accuracy of 95% for partially damaged documents. He and Schomaker [10] presented a joint feature distribution to improve the identification performance. They proposed two joint features for writer identification: run lengths of LBP and cloud-of-line distribution. These features were evaluated using handwritten irregularly curved stroke documents, and promising results were obtained. Obaidullah and others [11] proposed a handwritten document recognition system for 11 official Indic scripts. Their dataset has been used in many script identification and verification applications. They also obtained benchmark results for script identification. Aubin and others [2] introduced a grapheme-based offline method for identify verification through an analysis of handwritten strokes. This approach emphasized the utilization of simple individual strokes. Principal component analysis and the discrete cosine transform were used to reduce the data dimensionality. In addition, they used a support vector machine (SVM) with k-fold cross-validation to improve performance, achieving 100% accuracy in identity verification.

Bahram [1] presented a method for writer identification based on a modified LBP and measurements of ink-trace width and shape letters from handwritten documents. The method was evaluated on eight established handwriting datasets, achieving the highest performance among various evaluated methods on six benchmark datasets: KHATT, CVL, Firemaker, BFL, CERUG-CN, and ICDAR2013. Adak and others [12] studied writer identification and verification using various handcrafted and automatically extracted features from a dataset consisting of samples from 100 writers with diverse handwriting styles. By employing multiple automatically extracted features in combination with neural networks, they achieved promising results, indicating the effectiveness of their approach. In a recent study, a fusion neural network [13] that combined a broad learning system [14] and long short-term memory network was developed to accurately predict the capacity and remaining useful life of lithium-ion batteries [15]. Overall, we found scarce research on writer verification using deep learning.

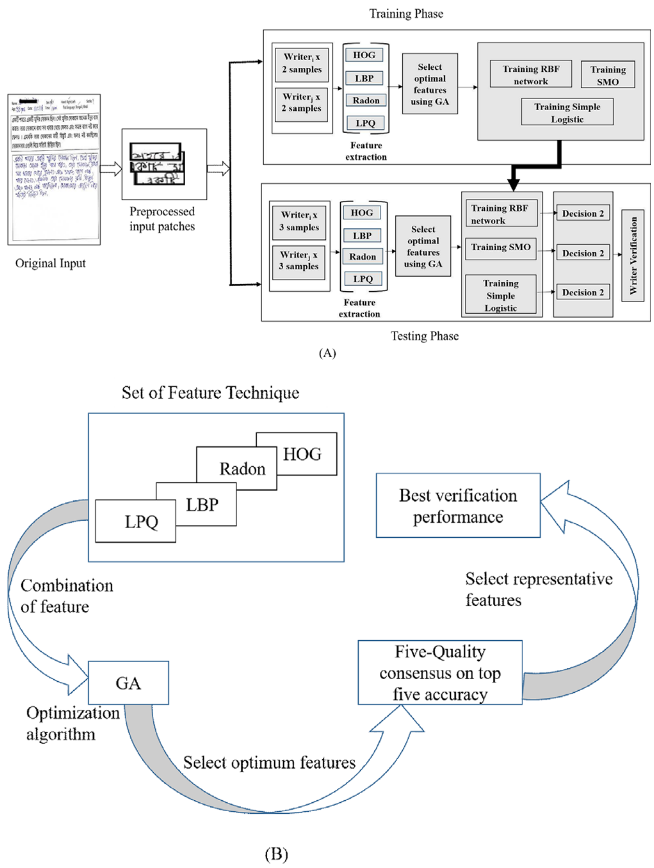


FIGURE 1 Diagrams of (A) proposed writer verification method and (B) writer verification procedure.

3 | PROPOSED WRITER VERIFICATION METHOD

The proposed method comprises (1) collection of handwriting samples, (2) data preprocessing, (3) extraction and combination of features, 4) GA-based feature selection, and (4) writer classification, as depicted in Figure 1.

In this study, we divided the samples from five writers into three disjoint training sets and two disjoint test sets. In Paul and others [16], we maintained a 3:2 splitting ratio of training and test samples over block- and page-level datasets. We chose this ratio to allocate a larger portion of the data for training, thereby ensuring adequate learning of complex patterns and performance improvement. The remaining data were used for testing to evaluate the generalization ability and performance on unseen data. For paired writers, nonoverlapping writer samples were randomly selected from the page and block levels. A GA was used to reduce the dimensionality of the features, obtaining the optimal number of selected features to enrich the results. This procedure was performed on both the block- and page-level datasets. Finally, a classifier based on these features was constructed using cost-sensitive learning with simple logistic regression, SMO, and a radial basis function (RBF) network.

3.1 | Dataset construction

We introduce an offline handwritten Bangla dataset called the Jadavpur University Deep Learning in Vision and Language Processing Bangla Language Writer Verification (JUDVLP-BLWVdb) dataset. Bangla is the most widely used language in various areas of Eastern India, but no publicly available benchmark dataset for writer verification is available. Our dataset was collected from 100 native Bengali writers at the document level, where each writer wrote the same content five times with a standard ballpen using blue or black ink. Faculty members of the Computer Science and Engineering Department of Jadavpur University assisted in data collection. To account for variations in the writers' handwriting styles at different times, we collected multiple samples from

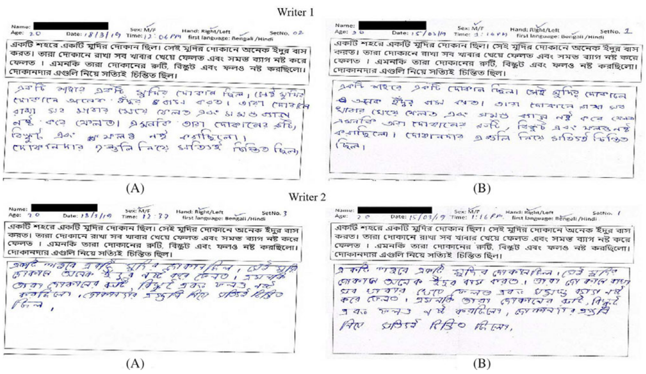


FIGURE 2 Sample document images of two different writers taken from our constructed dataset for writer verification.

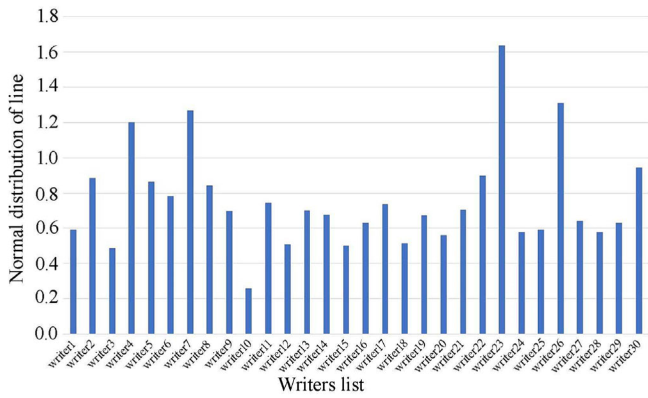


FIGURE 3 Normal distribution of sample text image line.

each writer. Figure 2 shows samples from the JUDVLP-BLWVdb dataset, which also included information such as age, sex, date, time, and vernacular language.

3.2 | Data preprocessing

All handwritten text pages were digitized using an HP LaserJet Pro M1136 scanner at 8-bit grayscale and 300 dots per inch to obtain 2481 × 3507-pixel text images. We labeled the digital text image data according to a writer identification number, which was set during preprocessing. In addition, processing involved converting the digital color image into a grayscale image, in which the pixel values were adjusted between 0 and 255 to reflect their intensities in the color space. GIMP was used to extract handwritten text from each grayscale image. Before cropping the handwritten text, we corrected the grayscale skew. A grayscale image was automatically converted into a binary image using a threshold derived from Otsu's method [17]. We then applied the minimum bounding box algorithm [11] to the binary image to find the minimum area-enclosing rectangle. After extracting the minimum bounding box of the text image, we applied a Gaussian distribution to determine the line distribution, as illustrated in Figure 3. If we obtained a value of seven using this method, the text image was divided into 7 × 7 rows and columns. We then improved the quality of the collected samples by applying the preprocessing steps depicted in Figure 4.

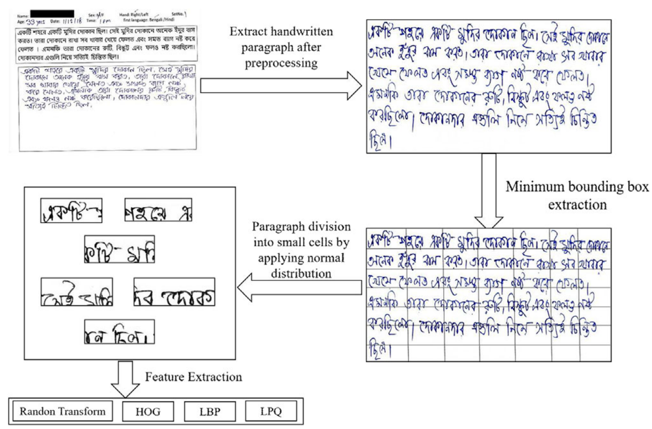


FIGURE 4 Diagram of data preprocessing and feature extraction for training and test phases in the proposed method.

3.3 | Texture information from handwriting image

Writer verification is a binary classification problem, in which the corresponding method must determine whether a handwritten document was written by the same or a different writer. To solve this problem, we extracted different texture-based handcrafted features, such as the Radon transform (FD_{rt}), HOG (FD_h), LBP (FD_{lb}), and LPQ (FD_{lq}). In addition to texture-based features, additional features were automatically extracted using deep learning.

3.3.1 | Radon transform feature FD_{rt}

The Radon transform feature is calculated by determining the projection of an image from a specific direction. Different angles are generated by rotating the center of the image. The projection is performed using (1) for a two-dimensional binary image $f(x,y)$ [18]:

$$R(r,\theta)[F(x,y)] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta \; (r-\theta x \, x\cos\theta - y\sin\theta dxdy) \tag{1}$$

where θ is the angle. We compute the maximum value of Radon transform angle θ that satisfies $1 \le \theta \ge 180$. The dimensions of the feature are $180 - D$.

3.3.2 | HOG feature FD_h

HOG [16] is a representative feature in pattern recognition. The image gradient is calculated for each pixel. This feature is expressed as

$$dp = I(p,q) - I(p-1,q) \tag{2}$$

$$dq = I(p,q) - I(p,q-1) \tag{3}$$

Next, we calculate gradient magnitude v and orientation θ as follows:

$$v = \sqrt{p^2+q^2}, \theta = \tan^{-1}\left(\frac{dp}{dq}\right) \tag{4}$$

Each image block is resized to 64 × 64 pixels before obtaining the HOG feature. We compute the feature for small patches of handwritten text using 3 × 3 HOG windows. The feature dimension is 81. We use nine rectangular blocks per image and 32 bin histograms per block. To create the resulting features per image, we concatenate nine histograms, each containing nine bins, resulting in an 81-dimensional feature. Figure 5 shows the output angle and magnitude from an input image.

3.3.3 | LBP feature FD_{lb}

We propose a curvature-free feature for writer verification. Specifically, we apply a grayscale and rotation-invariant texture operator based on LBPs [19]. The operator measures the spatial structure of the local image texture. The operator then computes the LBP transformation of the input image. We normalize each LBP block (block size of 32 × 32) histogram using the L1 norm. The obtained feature dimension is 236. Figure 6 shows the LBP feature obtained from an input image.

3.3.4 | LPQ feature FD_{lq}

We also consider the well-known LPQ method. The LPQ feature also measures the texture from handwritten text images. LPQ provides a texture feature that is robust to image blurring owing to its phase information and has been used in various pattern recognition tasks.

LBP and the retrieval of texture information from the histograms of LPQ labels computed within local regions are equivalent. Conventional LPQ operates by quantizing the phase of the Fourier transform within local neighborhoods, as illustrated in Figure 7. This process computes

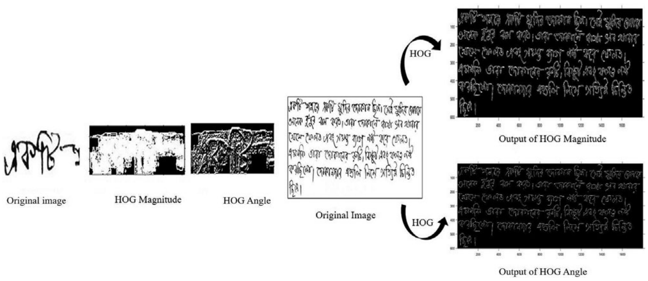


FIGURE 5 Output histogram of oriented gradients (HOG) feature from block and page images.

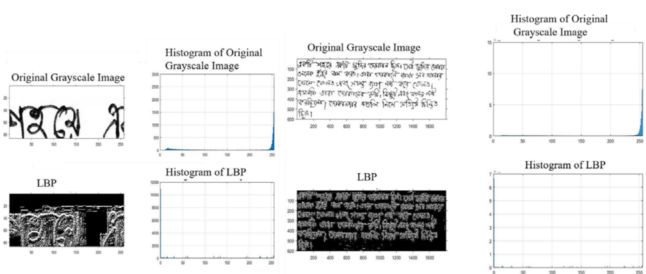


FIGURE 6 Output local binary pattern (LBP) feature from block and page images.

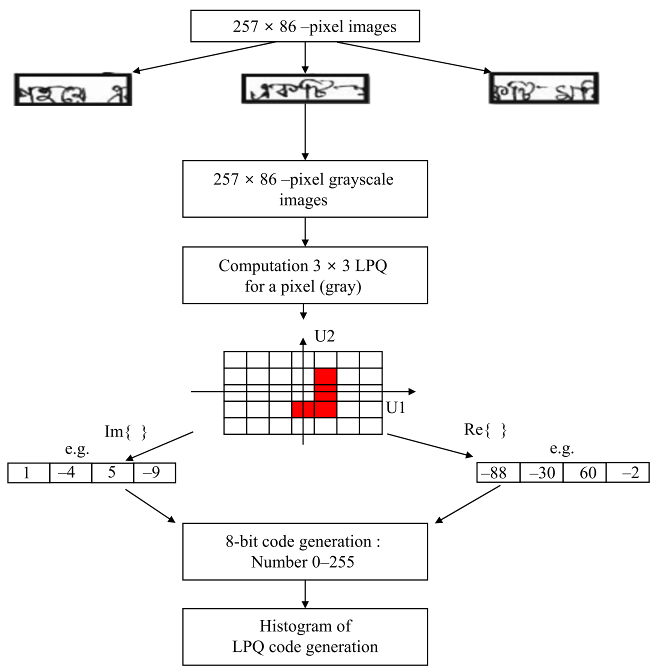


FIGURE 7 Diagram of calculation of local phase quantization (LPQ) feature.

the short-term Fourier transform within a rectangular $W \times W$ neighborhood denoted as M_i at every pixel i in image $f(i)$ as follows:

$$F(u,i) = \sum_{y \in N_i} f(i,y) e^{-j2\pi u^T y} = w_u^T f_i \quad (5)$$

where the short-term Fourier transform is implemented using two-dimensional convolution $f(i)^{-j2\pi u^T i}$ for all u. More details about LPQ are available in Tran and others [20]. The quantized integer coefficients range from 0 to 255 using a binary code and are accumulated in a 256-bin histogram, thus obtaining a 256-dimensional LPQ feature.

3.4 | Combination of features

Features FD_{rt}, FD_h, FD_{lb}, and FD_{lq} have 180, 81, 236, and 256 dimensions, respectively. Thus, the combined feature dimension is given by

$$FD_{\text{total}} = FD_{rt} \cup FD_h \cup FD_{lb} \cup FD_{lq} = 753 \quad (6)$$

3.5 | Feature selection using GA

GAs can be used for this dimensionality reduction. In our GA implementation [21], the number of genes represents the number of available features. The GA involves various processes, including chromosome encoding, population initialization, fitness (objective) function calculation, selection, and termination criteria. The objective function evaluates the quality and effectiveness of each candidate feature subset by assigning a fitness score based on its performance in terms of an evaluation metric. Then, the GA iteratively improves the feature subsets over generations using selection, crossover, and mutation operations guided by the fitness scores. Filter selection assesses the relevance of features independent of the learning algorithm, whereas wrapper selection incorporates a learning algorithm to evaluate subsets of features. We adopt wrapper selection, which involves the integration of a learning algorithm for the evaluation of subsets of features. This approach allows training and testing the learning algorithm on different feature subsets, enabling the identification of the optimal subset that maximizes performance. A detailed implementation of the GA is available in Babatunde [22]. The GA parameters are listed in Table 1.

TABLE 1 GA parameters for optimal selection from 753 features.

GA parameter	Value
Population size	50
Chromosome length	753
Population type	bitstrings
Fitness function	SVM
Number of generations	100
Crossover probability	0.8
Mutation probability	0.1
Elite count	2

GA (genetic algorithm).

3.6 | Automatically extracted features

We also use a deep convolutional neural network and features automatically extracted from images for pattern recognition, specifically for distinguishing handwriting text images. The AlexNet model is used owing to its advantages over the similar LeNet architecture regarding aspects such as performance and robustness. The simple convolutional architecture of AlexNet [23] automatically

TABLE 2 AlexNet parameters used for writer verification.

Parameter	Value
Number of classes	2
Data augmentation	Resizing to 224 × 224
Number of epochs	50
Batch size	1
Number of workers	1
Loss function	Negative log-likelihood
Optimizer	Adam
Classification function	Softmax

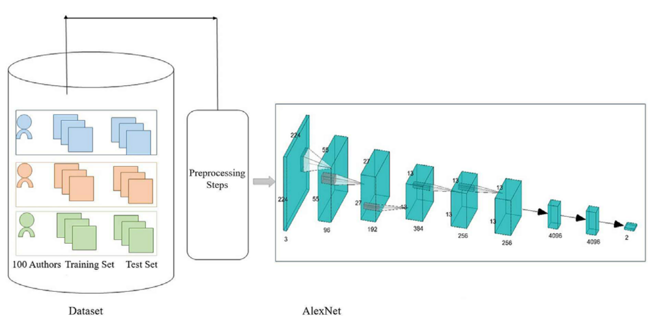


FIGURE 8 Architecture of deep learning model with automatic feature extraction.

extracts features and consists of a front part for feature extraction and a rear part for classification. Preprocessing is applied to resize the input image to 224 × 224 pixels and horizontally flip the image to introduce variability. The text is divided into block images, and data augmentation is performed using 25° and 35° flip operations. AlexNet for classification has multiple hidden layers, including five convolutional layers, three max-pooling layers, two normalization layers, two fully connected layers, and one softmax layer. Table 2 lists the parameters of AlexNet, and Figure 8 shows its architecture.

3.7 | Proposed classifier for writer verification

In the proposed method for offline text-dependent writer verification, we use WEKA tools [24] for classifying the obtained features. We employ SMO for analysis using simple logistic regression and incorporate a RBF network into the analysis.

3.7.1 | SMO algorithm

SMO [25] is an optimization algorithm that can easily handle large feature vectors extracted from handwritten images and extensive training and test samples. We use a polynomial kernel given by (7), where the polynomial kernel function is $K(x,y)$ and p is the degree of the polynomial. We tune the parameters of the SMO classifier for writer verification. The complexity parameter, C, is set to 1.0, and the batch size is set to 100.

$$K(x,y) = (x,y)^p \text{ or } K(x,y) = (x,y) + 1^p \tag{7}$$

3.7.2 | RBF network

A recent feedforward artificial neural network is based on the RBF [26]. The RBF network implements a normalized Gaussian RBF architecture. A random seed is used for k-means clustering of two clusters. A minimum standard deviation of 0.1 is specified for the clusters.

3.7.3 | Simple logistic regression

Simple logistic regression is a classification method [27] for recent developments using a linear model. For the simple logistic classifier, the maximum number of iterations is set to LogitBoost of 500. For small and large datasets, lower and higher values may be preferred, respectively. The beta parameter is used for weight trimming in LogitBoost.

3.8 | Experimental setup

The experiments were performed using MATLAB R2017b on a system equipped with an Intel i5-8250U core processor clocked at 1.60 GHz and with 8 GB RAM. For simulation, we used WEKA version 3.8, which provides various machine learning algorithms for ease of computation. We used our novel JUDVLP-BLWVdb dataset for the experiments. The dataset was distributed as follows: 500 pages from 100 writers, with five samples per writer. For training, three samples were selected (100 writers × 3 samples), and the remaining two samples were used for testing (100 writers × 2 samples). For verification, data from randomly selected writers with the same number of sample distributions were used during training and testing. For block-level training data, each writer had 147 block images (i.e., 49 blocks × 3 samples), and randomly selected writers had the same number of block images (Table 3). During training, we stored 29,400 block images (i.e., (147 + 147) × 100). During testing, 19,600 block images were stored (that is, (98 + 98) × 100 writers). Therefore, the distribution of page- and block-level data had a ratio of 3:2.

4 | EXPERIMENTAL RESULTS

To evaluate the proposed method for text-dependent writer verification, we conducted a comparative analysis using a feature-combination model. Table 4 lists the computational time and performance of the proposed method. The simple logistic classifier achieved average performances of 89.63% and 78.66% for 50-writer block

TABLE 3 Verification performance for n-fold cross-validation on entire dataset.

No. folds for cross-validation	Precision	Recall	F-measure	Accuracy (%)
3	94.10	94.06	94.05	93.08
4	94.40	94.38	94.38	94.37
5	94.57	94.60	94.56	94.55

TABLE 4 Performance for block- and page-level image datasets.

Classifier	50-writer accuracy (%)		100-writer accuracy (%)			
	Block image	Page image	Block image	Computation time (s)	Page image	Computation time (s)
Simple logistic	**89.63**	56.12	88.81	1.63	71.89	0.046
SMO	89.38	**78.66**	**89.76**	0.06	**81.02**	0.016
RBF network	78.84	68.22	78.38	6.25	71.89	0.027

Abbreviations: RBF, radial basis function; SMO, sequential minimal optimization.

and page images, and 89.76% and 81.02% for 100-writer block and page images, respectively. The performance for the page image dataset was lower than that for the block image dataset, which could be attributed to the small size of the training and test sets. Therefore, the GA for feature selection could not be applied to the page images of the JUDVLP-BLWVdb dataset. The SMO classifier performed better than the simple logistic and RBF network classifiers, and its accuracy increased with the number of writers for all classifiers.

To reduce the computational time and memory requirements, we reduced the dimensionality of the combined features. Table 5 lists the writer verification results. The dimension of the block image features was reduced from 753 to only half by using the GA. SMO achieved the best performance, with average accuracies of 94.06% and 94.05% for 100 and 50 writers, respectively. The simple logistic and RBF network classifiers provided the lowest accuracies. The results showed the same trend, as indicated in Figure 9. With increasing number of writers, the verification performance also improved.

To obtain the best results, we used the SMO classifier and five-quality consensus [28] by selecting five optimal features through five different runs of the GA. The results were obtained using the SMO classifier. Table 6 shows that a maximum writer verification accuracy of 94.54% with 724 features was achieved for four-quality consensus. Therefore, the GA produced the best writer verification performance using the SMO classifier. A single-quality consensus generated a maximum verification rate of 94.54% for the five best runs using the SMO classifier only. Figure 10 shows a set of predicted images, demonstrating both misclassification and correct classification results. Images originally belonging to writer 1 were incorrectly predicted as belonging to writer 2, whereas those belonging to writers 3 and 4 were correctly verified using the SMO classifier.

We also evaluated the writer verification methods using automatically extracted features and obtained the results for samples at the block and page levels, as listed in Table 7. A learning rate of 0.001 was used over

TABLE 5 Performance for block image dataset using GA.

Classifier	50-writer accuracy (%)	100-writer accuracy (%)
Simple logistic	93.56	93.71
SMO	**94.05**	**94.06**
RBF network	81.53	81.30

GA (genetic algorithm); RBF (radial basis function); SMO (sequential minimal optimization).

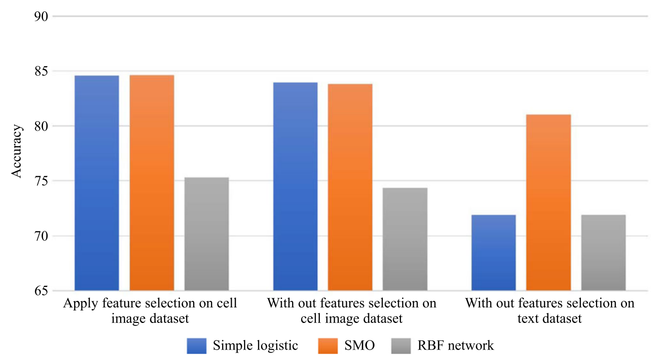


FIGURE 9 Verification accuracy for block- and page-level image datasets.

TABLE 6 Five-quality consensus results based on best five experimental results of features selected by GA using SMO classifier.

n for quality consensus	Optimal no. features	100-writer verification accuracy (%)
1	724	**94.54**
2	165	91.69
3	92	90.53
4	49	88.19
5	22	79.94

Abbreviations: GA, genetic algorithm; SMO, sequential minimal optimization.

100 epochs. The results for the JUDVLP-BLWVdb dataset at the block level were better than those at the page level. The average computation time of AlexNet at the block level was 11.66 h for 100 epochs, whereas that at the page level was 5 h for 100 epochs. Owing to the limited number of handwritten samples per writer, the results did not improve when using automatically extracted features compared with handcrafted features.

The results for various state-of-the-art methods and benchmark datasets are listed in Tables 8 and 9. Table 8 lists the writer verification performance when applied to different standard writer verification datasets. Table 9 shows the performances of various models on the JUDVLP-BLWVdb dataset. The scale-invariant feature transform was evaluated using GA dimensionality reduction, resulting in a reduction of less than 50% at both the block and page levels of the JUDVLP-BLWVdb dataset. The average accuracies obtained by using the scale-invariant feature transform were 60.03% and 57.33% at the block and page levels, respectively. Additionally, the graphemes feature was evaluated using GA dimensionality reduction at the page level of the JUDVLP-BLWVdb dataset, yielding an average accuracy of 58.65%. The computation time required for training the long short-term memory network at the block level was 9.83 h after 130 training epochs at a learning rate of 0.05 and batch size of 128. On average, the long short-term memory network achieved an accuracy of 73.64%.

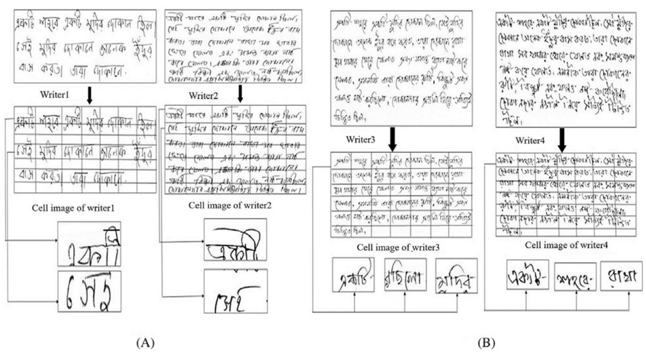


FIGURE 10 (A) Incorrectly and (B) correctly classified samples using sequential minimal optimization (SMO) classifier.

We performed n-fold cross-validation on the entire dataset. As in our data, we had five sets of samples per writer and performed from three to five folds using the SMO classifier. The SMO classifier was selected given its superiority over other classifiers, achieving 94.55% (94.37%) accuracy in fivefold (fourfold) cross-validation. Table 3 lists the results of n-fold cross-validation using the SMO classifier. The results were comparable with those obtained in our training and test model.

4.1 | Statistical significance test

We evaluated two approaches in our experiments. Approach 1 incorporated GA-based feature selection

TABLE 7 Average writer verification accuracy obtained using automatically extracted features on block- and text-image JUDVLP-BLWVdb datasets.

Dataset	50-writer accuracy (%)	100-writer accuracy (%)	Computation time (h)
Block level	**80.68**	**78.58**	11.66
Page level	69.08	68.00	5

TABLE 8 Writer verification performance of proposed method on different standard datasets.

Sl.	Dataset/source	Method	Accuracy (%)
1.	Halder and others [29]	Proposed	89.85
2.	Brazilian Forensic Letter [30]	Proposed method	93.26
3.	JUDVLP-BLWVdb	Proposed	**94.54**

TABLE 9 Writer verification performance for different methods on JUDVLP-BLWVdb dataset.

Sl.	Dataset	Method	Accuracy (%)
1.	JUDVLP-BLWVdb	Halder et al. [8]	63.68
2.	JUDVLP-BLWVdb	Hanusiak et al. [30]	80.44
3.	JUDVLP-BLWVdb	Aubin et al. [31]	85.29

with the SMO classifier and achieved higher average results than the other two classifiers. Approach 2 used the SMO classifier without GA-based feature selection. Both approaches were evaluated regarding the average results obtained from 100 writers. We applied these models to the block image data and obtained accuracies of 84.61% and 83.81% for approaches 1 and 2, respectively.

To demonstrate the significance of the performance gain resulting from GA-based feature selection, we performed a McNemar test. The estimated test value was greater than the critical value of 3.84 in the 95% confidence interval. Therefore, we rejected the null hypothesis and concluded that the two approaches had statistically significant differences in their performance. Differences in solutions between methods were considered statistically significant at a significance level of 0.05.

5 | CONCLUSIONS

We propose a writer verification method that showed promising results for page-level writer verification using a combination of Radon transform, HOG, LBP, and LPQ features. We applied the GA with SVM (used as the fitness function) for feature selection, which reduced the number of features by approximately 50% and achieved a maximum verification accuracy of 94.54% on our newly developed Bangla dataset. We found that our handcrafted feature combination was more robust than the automatically extracted features owing to limitations in the dataset. SVM outperformed commonly used classifiers such as a simple logistic regression and RBF network. The proposed method achieved a better performance using block images at the paragraph level. In future work, we plan to improve the verification performance at the line, word, and character levels. In addition, grapheme-based features will be incorporated to enhance the performance of the proposed method. Furthermore, more data will be collected, and data augmentation will be explored to achieve better results with deep learning models. Graphology has made significant progress in English handwriting analysis. The possibility of using graphological features for the analysis of Bangla scripts after employing optical character recognition will also be explored.

CONFLICT OF INTEREST STATEMENT

The authors declare no potential conflicts of interest.

ORCID

Jaya Paul https://orcid.org/0009-0004-6303-2939
Kaushik Roy https://orcid.org/0000-0002-3360-7576

REFERENCES

1. T. Bahram, *A texture-based approach for offline writer identification*, J. King Saud Univers. - Comput. Inform. Sci. **34** (2022), no. 8, Part A, 5204–5222.
2. V. Aubin, M. Mora, and M. Santos Peas, *Off-line writer verification based on simple graphemes*, Pattern Recogn. **79** (2018), 414–426.
3. J. A. Lewis, *Forensic document examination: fundamentals and current trends*, 2014, pp. 1–212.
4. R. Fernandez-de Sevilla, F. Alonso-Fernandez, J. Fierrez, and J. Ortega-Garcia, *Forensic writer identification using allographic features*, (23th International Conference on frontiers in Handwriting Recognition, Kolkata, India), 2010, pp. 308–313.
5. F. J. Zareen and S. Jabin, *Authentic mobile-biometric signature verification system*, IET Biometrics **5** (2016), no. 1, 13–19, DOI 10.1049/iet-bmt.2015.0017.
6. V. Aubin, M. Mora, and M. Santos, *A new approach for writer verification based on segments of handwritten graphemes*, Logic J. IGPL **30** (2022), no. 6, 965–978.
7. A. Bensefia and H. Tamimi, *Validity of handwriting in biometric systems*, (Proceedings of the International Conference on Pattern Recognition and Artificial Intelligence, Union, NJ, USA), 2018, pp. 5–10.
8. C. Halder, S. M. Obaidullah, J. Paul, and K. Roy, *Writer verification on Bangla handwritten characters*, Vol. **2**, Springer, India, 2016.
9. M. A. Khan, N. Mohammad, G. B. Brahim, A. Bashar, and G. Latif, *Writer verification of partially damaged handwritten arabic documents based on individual character shapes*, Peer J. Comput. Sci. **8** (2022), e955.
10. S. He and L. Schomaker, *Writer identification using curvature-free features*, Pattern Recogn. **63** (2017), no. C, 451–464.
11. M. S. Obaidullah, C. Halder, C. K. Santosh, N. Das, and K. Roy, *Phdindic_11:page-level handwritten document image dataset of 11 official indic scripts for script identification*, Multimed. Tools Appl. **77** (2018), 1643–1678.
12. C. Adak, B. B. Chaudhuri, and M. Blumenstein, *An empirical study on writer identification and verification from intra-variable individual handwriting*, IEEE Access **7** (2019), 24738–24758.
13. S. Zhao, C. Zhang, and Y. Wang, *Lithium-ion battery capacity and remaining useful life prediction using board learning system and long short-term memory neural network*, J. Energy Storage **52** (2022), 104901, DOI 10.1016/j.est.2022.104901.
14. C. Zhang, S. Zhao, Z. Yang, and Y. Chen, *A reliable data-driven state-of-health estimation model for lithium-ion batteries in electric vehicles*, Front. Energy Res. **10** (2022), 1013800.
15. C. Zhang, S. Zhao, Z. Yang, and Y. He, *A multi-fault diagnosis method for lithium-ion battery pack using curvilinear manhattan distance evaluation and voltage difference analysis*, J. Energy Storage **67** (2023), 107575, DOI 10.1016/j.est.2023.107575.
16. J. Paul, A. Sarkar, N. Das, and K. Roy, HOG and LBP based writer verification, in *Proceedings of international conference on frontiers in computing and systems*, D. Bhattacharjee, D. K. Kole, N. Dey, S. Basu, and D. Plewczynski (eds), Springer Singapore, 2021, pp. 3–12, DOI 10.1007/978-981-15-7834-2_1.
17. J. Miller, R. Patterson, D. Gantz, C. Saunders, M. Walch, and J. Buscaglia, *A set of handwriting features for use in automated writer identification*, J. Forensic Sci. **62** (2017), 722–734.

18. S. Bilan, R. Motornyuk, S. Bilan, and O. Galan, *User identification using images of the handwritten characters based on cellular automata and radon transform*, Biometric Identif. Technol. Based Modern Data Mining Methods **2021** (2021), 91–103.
19. S. Karanwal, *Robust local binary pattern for face recognition in different challenges*, Multimed. Tools Appl. **81** (2022), no. 20, 29405–29421.
20. C.-K. Tran, P. Khamphoui, et al., *Face recognition technology using the fusion of local descriptors*, Ann. Comput. Sci. Inform. Syst. **34** (2022), 227–231.
21. W. Siedlecki and J. Sklansky, *A note on genetic algorithms for large-scale feature selection*, Pattern Recogn. Lett. **10** (1989), no. 5, 335–347.
22. O. Babatunde, *Zernike moments and genetic algorithm: tutorial and application*, British J. Math. Comput. Sci. **4** (2014), 2217–2236.
23. S. Lu, Z. Lu, and Y.-D. Zhang, *Pathological brain detection based on AlexNet and transfer learning*, J. Comput. Sci. **30** (2019), 41–47.
24. A. Jain, P. Flynn, and A. Ross, *Handbook of biometrics*, Springer Science & Business Media, 2008.
25. H. Hassen and S. Al-Maadeed, *Arabic handwriting recognition using sequential minimal optimization*, (1st International Workshop on Arabic Script Analysis and Recognition, Nancy, France), 2017, pp. 79–84.
26. B. Zebardast and I. Maleki, *A new radial basis function artificial neural network based recognition for kurdish manuscript*, Int. J. Appl. Evol. Comput. (IJAEC) **4** (2013), no. 4, 72–87.
27. J. Friedman, T. Hastie, and R. Tibshirani, *Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors)*, Ann. Statist. **28** (2000), no. 2, 337–407.
28. 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*, Appl. Soft Comput. **12** (2012), no. 5, 1592–1606.
29. C. Halder and K. Roy, *Individuality of isolated bangla characters*, (International Conference on Devices, Circuits and Communications, Ranchi, India), 2014, pp. 1–6.
30. R. Hanusiak, L. Soares de Oliveira, E. Justino, and R. Sabourin, *Writer verification using texture-based features*, Document Anal. Recog. - IJDAR **15** (2011), 1–14.
31. M. N. Abdi and M. Khemakhem, *A model-based approach to offline text-independent arabic writer identification and verification*, Pattern Recogn. **48** (2015), no. 5, 1890–1903.

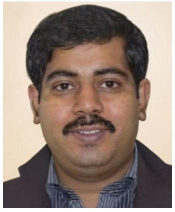
AUTHOR BIOGRAPHIES

Jaya Paul received her MTech degree from Jadavpur University, Kolkata, India, in 2008. Since February 2006, she has been working as an assistant professor in the Information Technology Department of the Government College of Engineering and Leather Technology, Kolkata, India. Her research interests include computer vision, machine learning, deep learning, and pattern recognition.

Kalpita Dutta received her BTech degree from Techno India, Saltlake, Kolkata, India, in 2013 and the MTech degree from the RCC Institute of Information Technology, Kolkata, India, in 2015. She received her PhD degree in Computer Science and Engineering from Jadavpur University, India, in 2023. She works on computer vision and machine learning.

Anasua Sarkar has been an assistant professor in the Department of Computer Science and Engineering, Jadavpur University, India, since 2016. She worked in universities and research laboratories in France, Sweden, and Denmark. She was the recipient of the EMMA-EPU fellowship from the European Union for pursuing doctoral research at University Bordeaux 1, France, in 2009. She also obtained a postdoctoral fellowship from Lund University, Sweden, to perform research on genetics in 2018. She is also a senior member of the Institute of Electrical and Electronics Engineers, USA, and a member of the International Rough Set Society. Her research interests include machine learning, pattern recognition, data science, bioinformatics, multiobjective optimization, biomedical signal processing, image processing, proteomics, epigenetics, and remote sensing. She is also interested in reading and technological advancements in learning.

Kaushik Roy received his BE degree in Computer Science and Engineering from NIT Silchar and the ME and PhD (engineering) degrees in Computer Science and Engineering from Jadavpur University in 1998, 2002, and 2008, respectively. He worked at institutes such as ISI-Kolkata and CDAC-Kolkata. He worked as an assistant professor at the Maulana Abul Kalam Azad University of Technology, India. He is currently working as a professor and head of the Department of Computer Science, West Bengal State University, Barasat, India. In 2004, he received his Young IT Professional Award from the Computer Society of India. His research interests include pattern recognition, document image processing, medical image analysis, online handwriting recognition, speech recognition, and audio signal processing. He is a member of ACM, a life member of IUPRAI (a unit of IAPR), the Computer Society of India, and a fellow of IETE.



Nibaran Das received his BTech degree in Computer Science and Engineering from Kalyani Government Engineering College, and the ME and PhD (engineering) degrees in Computer Science and Engineering from Jadavpur University in 2003, 2005, and 2012, respectively. He joined as an assistant professor at Jadavpur University in 2006. He is currently a professor in the Department of Computer Science, Jadavpur University. His research interests include optical character recognition, deep learning, artificial intelligence, pattern recognition, and image processing.

How to cite this article: J. Paul, K. Dutta, A. Sarkar, K. Roy, and N. Das, *Writer verification using feature selection based on genetic algorithm: A case study on handwritten Bangla dataset*, ETRI Journal **46** (2024), 648–659, DOI 10.4218/etrij.2023-0188.