



RESEARCH ARTICLE

OPEN ACCESS

AN EFFICIENT WORD-SPOTTING MODEL FOR GUJARATI LANGUAGE USING OPTIMIZATION DRIVEN DEEP LEARNING TECHNIQUE

Bijal Dalwadi^{1*}, Vatsal Shah²

¹Research Scholar, Gujarat Technological University, Ahmedabad, Gujarat, India

²Associate Professor, BVM Engineering College, VVNagar, Gujarat, India

¹<https://orcid.org/0000-0003-1805-531X>, ²<https://orcid.org/0000-0002-2570-6486>

E-mail: *bijaldalwadi7710@gmail.com, vatsal.shah@bvmengineering.ac.in

ARTICLE INFO

Article History

Received: March 09, 2026

Reviewed: April 09, 2026

Accepted: May 15, 2026

Published: September 30, 2026

Keywords:

Word Spotting,
Feature Extraction,
Convolutional Autoencoder,
Segmentation,
Feature Indexing.

ABSTRACT

Digitization of historical handwritten documents is highly preferable as it provides access to a wide range of documents to huge users. But, the information retrieval from such documents is a tough task as it requires full-text search. Word spotting is a process or retrieving the content from such documents based on a query image. This process results in a ranked list of words that are similar to the query word image. In several existing works, the discrimination between the feature vectors are not well understood. To resolve this issue, this paper deals with word spotting of Gujarati language. Initially the text document is pre-processed based on filtering, binarization and skew detection. Then, the pre-processed image is subjected to segmentation where line and word segmentation are performed to segment the words separately from the image. Then, the feature vectors are extracted from the word images using the proposed 3-layer deep convolutional autoencoder (CAE) and the model is further tuned using the improved elephant herding optimization (IEHO) algorithm. The feature vectors are then indexed using the geometric hashing algorithm and finally the matching is carried out based on Euclidean distance, Minkowski distance, hash similarity and Manhattan distance similarity measures. The framework is evaluated and compared with the existing models and the results proved the efficacy of the proposed model over the other existing models.



Copyright ©2026 by authors and Galileo Institute of Technology and Education of the Amazon (ITEGAM). This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

I. INTRODUCTION

South Asian repositories has vast collections of precious handwritten and printed documents. The digital format of these documents are available via electronic media forms [1]. But these collections are very huge, unstructured and identifying particular information is complex and time consuming process. The solution for this issue is a manual analysis or translation of these documents is huge process on the basis of labour, time, and cost [2]. Recently, due to the growth in communication and information technology, the handwritten or printed documents are protected, proceeded and accessed in digital format. Gujarati language is an official language of west part of India. It has 36 consonants and 11 vowels and it is belongs to Devnagari language. It is developed and originated in Gujarat and it is spoken by more than 60.4 million people of India [3] [4]. The Gujarati language has possesses a rich literary heritage and important role in maintaining the cultural identity [5]. The Gujarati character set is twice than that of English languages. IR (information retrieval) is a process for retrieving related information from a documents [6]. It is easy to retrieve information from digitized text document but retrieving information from document images are complex process. IR and efficient indexing are necessary requirement of any system that handles the digital content [7] [8]. There are two methods are utilized for spotting the document images. They are OCR (Optical Character Recognition) method and word spotting. For Gujarati scripts, OCR methodology is still in progress [9]. Further, OCR based approaches are not an appropriate approach for printed documents. This is because of (a) less quality documents (b) variation in writing style, (c) different writing style and (d) changes in writing word [10]. An example of the Gujarati script taken from historic document is displayed in Figure 1.

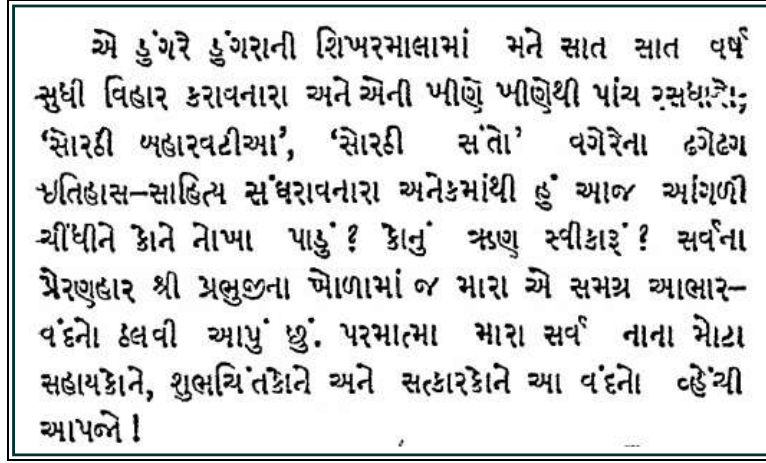


Figure 1: An example of the Gujarati script.

Source: Authors, (2026).

An alternative to OCR is word spotting in which the information identified is retrieved using matching the shape of a query words. It doesn't require any OCR process; handwritten or printed word spotting is the process that permits the user for searching for keywords in text [11]. That is method of retrieving entire cases of the given word from the document. Word spotting is a general process in IR processes in which the query image is matched with document image without explicitly recognize the characters [12]. Various word spotting approaches are depending on the pre-processing stage where a document images are segment into words. These words are compared with query image for detecting potential matches [13]. The use of word spotting is mainly focused in case of historical document images since they are poor quality and variation in writing style. Word spotting identifies a particular keyword in image and compared a features which are extracted from word image [14] [15].

1.1 MOTIVATION

Word spotting is one of the topic of study in recent times as there is a vast number of audience who are interested in acquiring information from historic documents. Retrieving contents from those documents is a tiring task and requires deep knowledge and efficient models. Word spotting models in the existing literatures are capable of acquiring the words from the documents but the problem with the existing models is that those models are computationally complex. Moreover, it is nearly impossible to search for the keywords in the digitized documents without the intervention of any recognition models. For a given query image, the recognition system finds the exact match for the query image by searching in the text documents. The existing approaches follow different segmentation processes and the segmentation process based on pages is found to generate different and infeasible outputs. On the other hand, several approaches follow segmentation-free models but these models take longer time to locate the words in the regions. Also, the computational cost of these models are high. Motivated by all these facts, this paper proposes an effective approach for word spotting based on discriminative feature extraction and segmentation steps.

1.2 CONTRIBUTION

The major contributions of the proposed work are as follows:

- The historic input images used for word recognition are affected with several factors. These factors diminish the overall accuracy rate in the detection process and may provide infeasible results. To overcome such problems, the proposed work initially pre-processed the input images based on filtering, binarization and skew detection processes.
- An effective line and word segmentation step is followed in the proposed work to differentiate each word in a text document so that the words can be easily matched with the query images.
- The feature vectors play a major role in the word recognition problems and based on the extracted features, the words are spotted from the document. Thus, an effective feature extraction step with the preservation of discriminative strength of the features is carried out in the proposed work.
- The extracted features are indexed to make the matching process easier and finally matching is carried out based on four different similarity measures to evaluate the entire approach.

1.3 PAPER ORGANIZATION

The remainder of the paper is organized as follows: section 2 presents a literature survey of the existing works related to word spotting, section 3 covers the proposed methodology with mathematical formulations and architectures, section 4 presents the results and discussion and section 5 concludes the paper.

II. RELATED WORK

Several research works related to word spotting in different languages are proposed in literature. Some of the most popular and effective ones are reviewed under this section. Majumder et al. [16] introduced word searching model which has two phase searching processes. In the first phase, the target word images were extracted from document and in the next phase robustness was increased. Initially, in pre-processing lengthy words which were not match with target images were removed.

Then, rule based approaches were used for pruning of images and DTW (dynamic time warping) with voting approach was used to find the filtered words are same as the query word images. The method was tested on Qatar University Writer Identification dataset and obtained better results. Cheikhrouhou et al. [17] presented a DL model end to end multi task for word spotting and identification of written script in multi languages. Text line image is taken as input and provides two results like position of keyword and script. Initially, local features were extracted from CNN's intermediate layer and global features were extracted from Bi-LSTM. Finally these two features were integrated into single representation by CBP (Compact Bilinear Pooling). The identification of script outcomes was provided in word spotting module for eliminating irrelevant characters scripts and carry out decoding phase. Farooqui et al. [18] proposed a model for handwritten word spotting of Urdu language. This work exploited the usage of randomly data generation for word spotting process. The system was trained on randomly varying data, hence it doesn't needed lengthy transformation and correcting processes. Initially, for image generation GAN (Generative Adversarial Network) and geometric transformation were used. Then, HoG (histogram of oriented gradients) features were extracted and the LSTM network was used for classification process. This model achieved better recognition rate of 89.9%. Das and Mandal [19] presented the segmentation free word spotting model for historical Bangla document on the basis of WKS (Wave Kernel Signature). Initially, document page and query image were smoothened and SIFT (Scale-invariant feature transform) detector was used for obtaining the keypoints.

The window was mounted at centre at every keypoint for obtaining the descriptor of WKS. This descriptors represented the mean probability of computing the quantum particle at the particular position on the basis of quantum energy. This model was verified on three datasets and performance was validated by standard measures. Kundu et al. [20] proposed learning free word spotting model followed with QBE (query by example) for handwritten documents. This model undergoes the stages like pre-processing, division of vertical zone, extracting feature and matching. Initially, the noise was removed and the vertical zone was used for splitting word image to different zones. HT (Hough transform) was used for extracting features and finally the images were compared with word images and finally metrics on the basis of DTW was used for estimating the similarity score. From the survey conducted, it has been identified that the existing methodologies are unable to completely meet the requirements of the users. Some of the methodologies in word recognition are OCR-based and are unable to provide effective results as it is unsuitable for printed documents. On the other hand, certain methodologies are formulated based on rules but the accuracy attained in recognition is not up to the desired level. The existing mechanisms based on deep learning are found to provide better results with higher accuracy. But the major problem is that there is a big lack of word recognition models in literature for Gujarati language. To deal with this issue in an effective manner, this article presents a word spotting model based on deep learning for Gujarati language.

III. PROPOSED METHODOLOGY

In this work, a new and effective model for word spotting in Gujarati language is proposed to help the users in recognition of the words from historical documents. There are five major steps in the proposed work such as pre-processing, segmentation, feature extraction, feature indexing and matching. At the initial pre-processing stage, filtering technique is applied to remove the noise and distortions and to enhance the image quality. Then, the images are binarized to convert them into black and white images. Further, skew detection is carried out to check the deviation in the words. In the second stage, line and word segmentation is performed to segment the words from the input. After the segmentation stage, the feature vectors are extracted using the 3-layer deep convolutional autoencoder (CAE) model. The optimal weights for fine-tuning the model are chosen by the improved elephant herding optimization (IEHO) algorithm. Then, the extracted feature vectors are indexed using the geometric hashing indexing technique. Finally, matching is conducted between the test image and query image in the dataset based on four similarity measures such as Euclidean distance, minkowski distance, Manhattan distance and hash similarity. The architecture of the proposed framework is displayed in Figure 2.

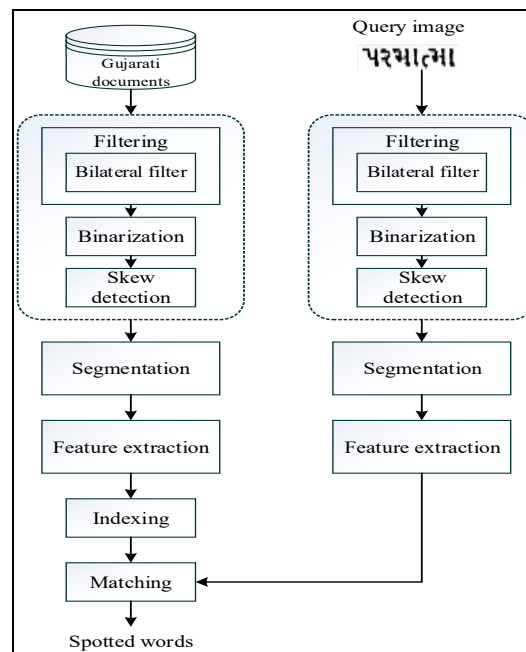


Figure 2: Architecture of the proposed word spotting model.

Source: Authors, (2026).

III.1 PRE-PROCESSING:

This is the initial stage of the proposed work where the images subjected for word spotting are pre-processed for quality enhancement. Since the images are from the historical documents, these images are affected by noise and distortions while scanning or conversion and transmissions. To enhance the accuracy in word spotting, it is important to remove these noises and enhancing the quality of the image. There are three main steps in pre-processing as follows:

- Filtering
- Binarization
- Skew detection

Step 1: Filtering

The primary step of pre-processing is filtering where the images are smoothed and the quality is enhanced. In the proposed work, the bilateral filter [21] is applied for filtering as this filter is capable of smoothing the image with the preservation of edge information. It is a nonlinear, simple and non-iterative filter that is capable of removing the noises from the images by retaining the important information. In this filtering process, the pixel values are averaged with the weight values that are computed based on the radiometric distance between the pixels. To deal with the intensity of the image, the range filtering is considered for computations. The overall expression of the bilateral filter is provided as the product of the range filter and the domain filter. The mathematical formulation of bilateral filter for a pixel at p can be expressed as follows:

$$\bar{I}(p) = \frac{1}{\eta} \sum_{q \in N(p)} \exp\left(\frac{-q - p^2}{2\sigma_s^2}\right) \exp\left(\frac{-|I(q) - I(p)|^2}{2\sigma_i^2}\right) I(q) \quad (1)$$

Where, $N(p)$ indicates the spatial neighborhood around p , σ_s indicates the control parameter referring to the decay of weight value in spatial domain, σ_i indicates the control parameter referring to the decay of weight value in intensity domain, q indicates a neighborhood location and η is the normalization constant which can be mathematically expressed as follows:

$$\eta = \sum_{q \in N(p)} \exp\left(\frac{-q - p^2}{2\sigma_s^2}\right) \exp\left(\frac{|I(q) - I(p)|^2}{2\sigma_i^2}\right) \quad (2)$$

In the proposed work, the window size of the bilateral filter is chosen based on the image size and the value for the control parameters are assigned based on the estimated noise level in the image. This filtering step results in a noiseless and smoothened image with better quality. The output of this step is then passed to the binarization process.

Step 2: Binarization

Binarization is the process converting an image into a dual-colored or bi-level image i.e. black and white image. The major reason for performing binarization is that this process enhances the performance of the word spotting technique and results in better output. The ultimate aim of binarization is that it segments the document into foreground text and background image. The proposed approach follows the local maxima and minima [22] approach for binarization. This method depends upon the contrast of the image of local minimum and maximum to convert the image into black and white image. The effect of variations in the image background is compensated using a normalization factor. The contrast of the image can be calculated as follows:

$$C(j, k) = \frac{I_{\max}(j, k) - I_{\min}(j, k)}{I_{\max}(j, k) - I_{\min}(j, k) + \delta} \quad (3)$$

Where, $I_{\min}(j, k)$ and $I_{\max}(j, k)$ are the minimum and maximum image intensities located within a local neighborhood window and is an infinitely small positive number that is added when the local maximum becomes zero. The numerator in equation (3) captures the local image difference and the denominator acts as the normalization factor that deals with the effect of variations in brightness and image contrast. The output of this process is then sent to the skew detection process to diminish the deviation between the words.

Step 3: Skew detection

Document skew is a type of distortion that arises while scanning or copying a document. The orientation of the lines in the document are affected and are termed to be skewed. This problem often occurs while automatically scanning bulk amount of documents and is generally unavoidable. Consider $I(p, q)$ indicate the word image with 1s indicating foreground pixels and 0s indicating background pixels. Let $q_{\max}$ and $p_{\max}$ indicate the height and width of the image. When the image exhibits skew, there will be changes in the left and right word projections. The global maxima of the left and right projections are computed for $q = q_l$ and $q = q_r$. The values for q_{r1}, q_{r2}, q_{l1} and q_{l2} are computed as in [23]. The mathematical formulation to compute the skew angle is given as follows:

$$\theta = \tan^{-1} \left(\frac{q_{r1} + q_{r2} - q_{l1} - q_{l2}}{p_{\max}} \right) \quad (4)$$

This step results in an enhanced image where the slanting and deviated words are made clear. The output of the pre-processing step is then subjected for segmentation to acquire the line and words separately from the word images.

III.2 SEGMENTATION

Segmentation is one of the crucial step in the proposed word spotting system to obtain better output. In general, the segmentation step is to segment the words from the text document to spot the words based on the given query. The segmentation is carried out in the proposed work based on line and word segmentation. Initially, the lines are segmented from the text document and then the words are segmented from the lines.

Line segmentation: In case of line segmentation, there are three major steps involved as follows:

- Run length smearing
- Obtaining middle lines
- Candidate line segmentation

Step 1: Run length smearing

The initial step in line segmentation is run length smearing and the strength of the histogram is increased with this smoothing technique. The binary image is taken as the input and the length of the black run between two white pixels is calculated and if the run length is found to be less than five times of the thickness of a line in a character, then the black run is filled with white. This smoothing technique results in horizontal projection profiles corresponding to the considered text lines. An example of the run length smearing is displayed in Figure 3.

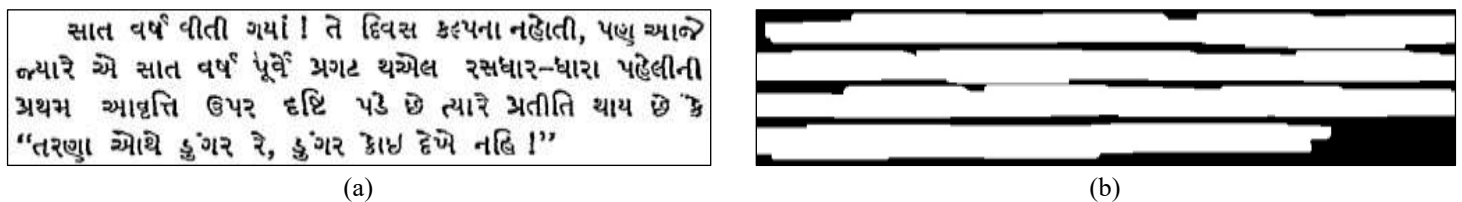


Figure 3: An example of run length smearing (a) original input image (b) horizontal profiles for the text lines.

Source: Authors, (2026).

Step 2: Obtaining middle lines

For a particular word image, the middle lines are obtained for line segmentation from the horizontal projection profiles. This is done by finding the highest peak line from all the projection profiles. Then, the middle point of the highest peak line is found and a vertical line from top to bottom is drawn at the middle point of the highest peak. For every peak of the histogram profiles, the middle line is obtained and from these lines, the middle points are acquired. Again horizontal lines are drawn based on the width of the histogram profiles from the middle point. For cases when the peak of histograms does not intercept with the vertical line, the distances between the middle lines are computed and the average value is considered. If the distance between the middle lines is found to be two times greater than that of the average value, then that region is considered to have one or more text lines. Thus, recursive segmentation is performed over that region. The procedure is repeated until the middle lines of the particular word image are obtained.

Step 3: Candidate line segmentation

Starting from the first histogram vertical scanning is done between the first and second middle lines to obtain two black pixels. These two pixels are considered to be the minimum points. The first black pixel line is considered to be first minimum and the second one as the second minimum. After this step, the vertical distances between the first middle line to the first minimum and second minimum and the maximum distance is considered between those two measures. The minimum point with maximum vertical distance is considered to be the line separator. These steps are repeated until all the line separators from the input word image are obtained. An illustration for line segmentation is presented in Figure 4.

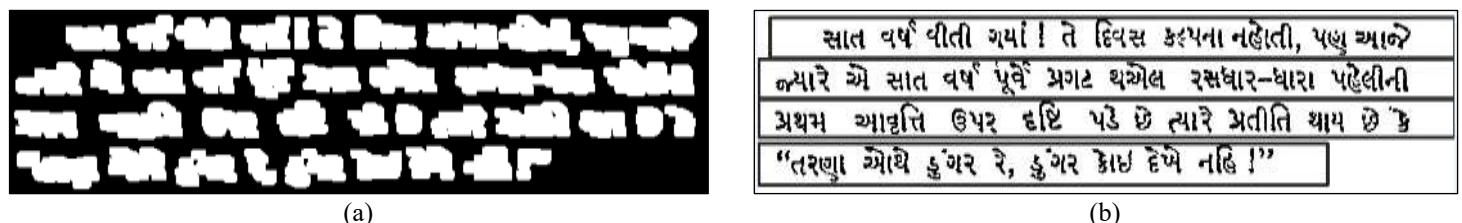


Figure 4: An illustration for line segmentation in the proposed work (a) vertical scanning procedure (b) segmented words.

Source: Authors, (2026).

Word segmentation

After line segmentation, the segmented lines are provided as the input to the word segmentation method. A single text line is first sent as input to the word segmentation process and it is scanned in a vertical manner. If a single vertical scan yields two or less black pixels, then it is indicated as 0 otherwise, indicated as the number of black pixels. By this way, a vertical projection profile is built. If there exist a run of k_1 consecutive 0s in the profile then, the boundary of the word is considered by the midpoint of the run. One-third of the height of the text line is taken as the value for k_1 . An illustration for word segmentation is displayed in Figure 5.

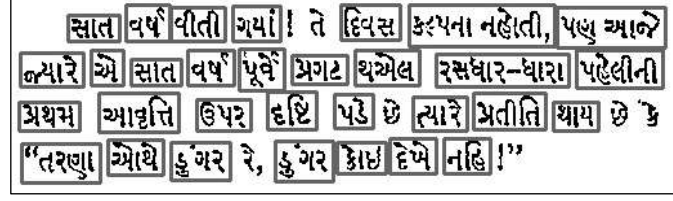


Figure 5: An illustration for word segmentation in the proposed work.
Source: Authors, (2026).

III.3 FEATURE EXTRACTION

The third stage of the proposed word spotting approach is feature extraction where the major features from the images are extracted. One of important requirement of the feature extraction step is that the discriminative nature of the features should be well understood. The application of autoencoders for feature extraction enhanced the process by maintaining the discriminative strength of the features. Also, the problem of dimensionality of features is reduced with the use of autoencoders. The proposed approach extracted the features from the word images using the 3-layer deep convolutional autoencoder (CAE) model. This model is composed of deep convolutional layers to extract the discriminative features from the images. The structure of the autoencoder used in the proposed model is displayed in Figure 6.

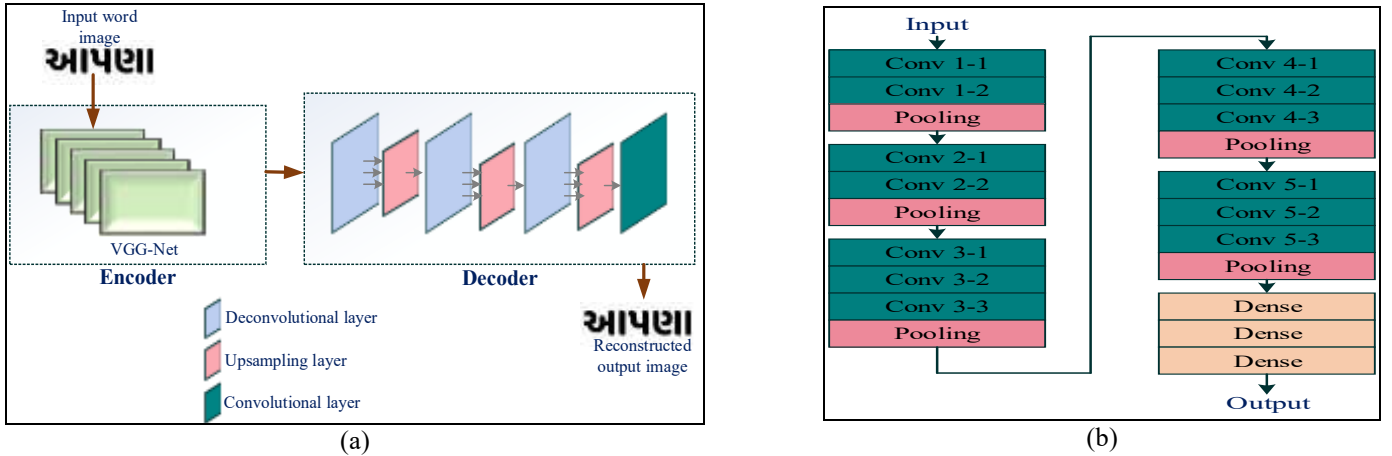


Figure 6: (a) Structure of the 3-layer deep CAE used in the proposed work (b) VGG-16 as an encoder for the proposed work.
Source: Authors, (2026).

The deep CAE model consists of two parts such as the encoder and decoder with convolutional and pooling layers to enhance the extraction process. The VGG-Net is used as the encoder that consists of the convolutional and pooling layers whereas, the decoder part consists of three deconvolutional layers and three upsampling layers. The use of the convolutional filters in the autoencoder is dedicated for feature extraction and the deconvolutional filters are the transpose of the convolutional layers. For the given input word image, the feature vectors are generated using the following convolution operation:

$$C_i = \sigma(\wp * f^n + b) \quad (5)$$

Where, σ indicates the sigmoid activation function, $\wp$ indicates the zero padding applied to the input to preserve the spatial resolution, f^n indicates the n^{th} convolutional filter of encoder and b is the bias value of the layer. The reconstruction of the extracted feature vector is carried out in the deconvolutional layer of the decoder given as follows:

$$D_i = \sigma(C_i * \tilde{f}^n + \tilde{b}) \quad (6)$$

Where, D_i indicates the reconstruction of the input, $\tilde{f}^n$ indicates the n^{th} convolutional filter in decoder and $\tilde{b}$ is the bias of the decoder.

The proposed CAE model for feature extraction is affected by error rate due to the addition of random parameters along with the input. It is thus mandatory to eradicate these errors to maintain higher accuracy. The error function of the CAE model can be formulated as follows:

$$L(\theta) = \sum_{i=1}^m (x_i - D_i)^2 \quad (7)$$

The proposed approach deals with the above formulated error rate with the help of a nature inspired optimization algorithm. The improved elephant herding optimization (IEHO) algorithm is implemented within the hidden layers of CAE model to enhance the extraction process.

III.3.1 IEHO Based Weight Tuning For CAE

The IEHO algorithm is employed in the proposed model for optimizing the fitness function which is given as follows:

$$F_t = \min (L(\theta)) \quad (8)$$

The improved optimization algorithm is inspired from the traditional elephant herding optimization (EHO) algorithm which is based on the behavior of elephant clans. The population of elephants is partitioned into a specific number of clans where a main female in the clan leads the group. This female (Matriarch) has the highest possibility of achieving the best fitness in the group. Along with this, the males in the clan separate from the clan when they reach adulthood. This results in a global search and results in worst or poor solutions with less fitness. The problem with the traditional algorithm is that the worst individual is replaced in a random manner. Also, the algorithm lacks proper exploitation process leading to slow convergence. Therefore, an improved version of the algorithm is introduced in [24] using the opposition based learning (OBL) and sine-cosine algorithm (SCA). The OBL computes and evaluates a candidate solution and its matching opposite solution and then identifies the best candidate solutions based on the fitness evaluation. The computation for matching opposite solution (O'_i) for the candidate solution (O_i) can be given as follows:

$$O'_i = u_b + l_b - O_i; \quad O_i \in [u_b, l_b] \quad (9)$$

Where, l_b and u_b indicates the lower and upper bounds of the solution space. The elephants in the search space exhibiting worst fitness are preferably selected and moved and then their new positions are computed using the new SCA based position updating formulation as follows:

$$x_{wr,ci} = \begin{cases} x_{wr,ci} + rnd1 * \sin(rnd2) * |rnd3 * x_{bs,ci} - x_{wr,ci}| & rnd4 < 0.5 \\ x_{wr,ci} + rnd1 * \cos(rnd2) * |rnd3 * x_{bs,ci} - x_{wr,ci}| & rnd4 \geq 0.5 \end{cases} \quad (10)$$

Where, $x_{wr,ci}$ indicates the worst individual in clan ci , $x_{bs,ci}$ indicates the best individual in clan ci , $rnd1, rnd2, rnd3$ and $rnd4$ are the random parameters to control the new position's region. The computations of the random parameters are as follows:

$$rnd1 = A - cr_i * \left(\frac{a}{mx_i} \right) \quad (11)$$

$$rnd2 = (2 * \pi) * rand() \quad (12)$$

$$rnd3 = 2 * rand() \quad (13)$$

$$rnd4 = rand() \quad (14)$$

Where, cr_i indicates current iteration, mx_i indicates maximum iteration, $rand()$ is a random weight function added for switching from sine to cosine. The clan is updated for every iteration using the traditional algorithm and the new position update formulation is utilized for worst solutions. By this way, the best solutions or the optimal weight values are identified for the layers to perform feature extraction thereby resulting in better accuracy.

III.4 FEATURE INDEXING

The next step to feature extraction is feature indexing where the features are indexed in the proposed work based on geometric hashing technique. This technique is employed to match the feature vectors generated with the features in the dataset.

The reason for choosing this technique is that this technique is capable of finding the matches even when the feature vectors have undergone certain changes or transformations. It is assumed that the feature vectors are randomly distributed subjected to probability density function. Based on this random process, a joint probability density function is defined as follows:

$$\int f(x(p, q), y(p, q))f(x_{v_1}, y_{v_1})f(x_{v_2}, y_{v_2}) \quad (15)$$

Where, (p, q) indicates the quantized solution to access the hash table. In case of similarity transformation, the feature points generated by the Gaussian random process and the integral evaluation yields the following:

$$f(p, q) = \frac{12}{\pi} \cdot \frac{1}{(4(p^2 + q^2) + 3)^2} \quad (16)$$

The above function is used to determine the indexes of the feature vectors in the hash table. Based on the numerical formulation, the indexes for the features in a random process can be easily obtained.

III.5 MATCHING

The final stage of the proposed work is matching of words where the obtained output word is matched with the input word based on some similarity measures. The proposed work used four major similarity measures to match the words and to determine the range of similarity. This step is predominant as it ensures the performance of the model in accurately spotting the words. The similarity measures used in the proposed work are Euclidean distance, Minkowski distance, hash similarity and Manhattan distance. The description and mathematical formulations of the four considered similarity measures are as follows:

III.5.1 Euclidean Distance

Euclidean distance is computed as the length or distance between two points in space. In the proposed work, the distance between two feature vectors are computed to determine the deviation of the spotted words. For any two feature vectors C and D , the Euclidean distance can be computed as follows:

$$EU(E, D) = \left(\sum_{i=1}^n |e_i - d_i|^2 \right)^{\frac{1}{2}} \quad (17)$$

This similarity tends to be significant when the value obtained is low. Lower value indicates that the deviation between the input word and the spotted word is low.

III.5.2 Minkowski Distance

Minkowski distance is another similarity or distance measure which can be considered as a generalization of the Euclidean and Manhattan distance measures. The mathematical formulation for Minkowski distance between two feature vectors can be given as follows:

$$MD(i) = \sum_{k=1}^n |Q_i(k) - S_i(k)| \quad (18)$$

Where, $Q_i(k)$ indicates the descriptor of query and $S_i(k)$ indicates the descriptor of word i .

III.5.3 Hash Similarity

Hash similarity is a similarity measure that is computed based on the hash values between the two vectors. If the images are more similar to each other, then the hash values taken between those two images will be the same.

III.5.4 Manhattan Distance

Manhattan distance or the taxicab distance is a distance measure used for measuring the absolute difference between two points. The mathematical formulation for Manhattan distance can be given as follows:

$$MH = |e_i - d_i| \quad (19)$$

All the above distance measures are utilized and the deviation between the spotted and original words are computed. Based on the similarity measures calculated, the words are ranked and the best word with rank 1 is obtained as the output.

IV. RESULTS AND DISCUSSIONS

This section covers the experimental evaluations and analysis of the proposed work. The proposed word spotting model is compared with several existing works to identify the performance improvement compared to others. The existing models chosen for comparison include voting method [16], multi-task learning [17], wave kernel signature [19] and Hough transform [20]. The detailed evaluations with performance metrics and analysis are presented in the upcoming sections.

IV.1 SIMULATION SETUP

The entire implementation of the work is carried out in the python platform. The sample of data in this research is Gujarati document images of historical literature, such as Saurashtra Ni Rasdhar (Part-1, Part-2, and Part-3) authored by Zaverchand Meghani. The number of scanned historical documents pages in the Gujarati language is high in these books. Based on these documents, over 90,000 words. In the word spotting experiments, a query set was created that comprised of 24 classes and 334 word images. Every type of class signifies a particular Gujarati word and they are aapana, avatar, bhagwan, Bhavnagar, dekhav, ghadki, gujarat, itihash, kathiyavadi, kundal, laptop, lekhako, maharaj, mahol, manan, nikhals, odhani, pardosh, samachar, sikharamala, ubharata, vadtal and vivek. Furthermore the 80% of the data for training and 20% of data for testing. The total number of convolutional layers are 3, total pooling layers are 3, initial population is 100, maximum iteration is 300, learning rate is 0.001, batch size is 32, epochs is 100, activation function is softmax and dropout rate is set to 0.2. The system configuration is as follows: the implementations are carried out in 3.20 GHz processor with 8 GB RAM running on 64-bit windows 10 operating system.

IV.2 PERFORMANCE METRICS

The proposed method is evaluated in terms of precision and recall. The mathematical formulations for the metrics are as follows:

$$\text{Pr} = \frac{|W_v \cap W_r|}{|W_r|} \quad (20)$$

$$\text{Re} = \frac{|W_v \cap W_r|}{|W_v|} \quad (21)$$

Where, W_v indicates the set of valid instances for a given search word and W_r indicates the set of retrieved instances.

IV.3 PERFORMANCE ANALYSIS

The proposed method is compared and analyzed with the existing methods under this section. The input text image is initially provided to the proposed framework where the pre-processing steps are carried out. The BF is initially applied and noise and distortions are removed. Then, binarization is carried out to convert the input image into a bi-level image. After this step, skew detection is done to maintain the orientation of the document. Then, line and word segmentation is carried out to separate lines and words from the document. After that, feature vectors are extracted from the word images using the 3-layer deep CAE model where the convolutional filters are employed for feature extraction. The feature extraction step is optimized using the IEHO algorithm. Then, the feature vectors are indexed and matching is carried out based on four major similarity measures. Based on the similarity values obtained, rank is assigned to each obtained word. The word with better rank is retrieved as the output for the given query image. The evaluations and illustrations are as follows:

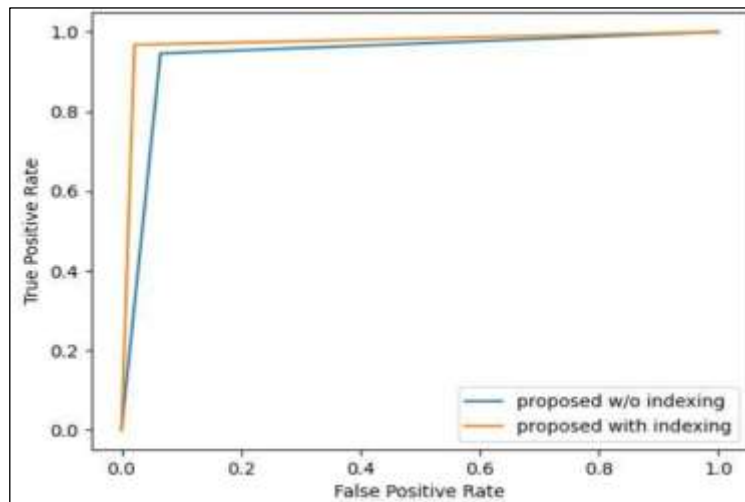


Figure 7: ROC curve for the proposed model (AUC with indexing = 97.282, AUC w/o indexing = 94.021).

Source: Authors, (2026).

Figure 7 presents the receiver operating characteristic (ROC) curve for the proposed model. From the curve, it is clear that the proposed model performed better in extracting the features compared to the other models. The CAE model employed for feature extraction extracted the discriminative features thereby resulting in effective spotting of the words. The true positive of the model is higher and the model provided better results. The overall area under the curve (AUC) value obtained by the proposed model is 97.282 proving that the model is ideal than most of the existing frameworks related to word spotting. The framework is also evaluated with the elimination of the indexing procedure to identify the contribution of indexing in word spotting. Compared to the curve of the proposed model, the curve of the proposed model without indexing seems to be low indicating lower AUC value. The overall AUC of the proposed model without indexing procedure is 94.021 which is lower than the original value. This analysis proves that the indexing is one of the important procedure in the proposed model to gain effective improvement in spotting the words from gujarati texts.

Table 1: Performance comparison of the proposed and existing methods.

Methods	Precision (%)	Recall (%)
Voting method [16]	13.11	47.79
Multi-task learning [17]	89.004	-
Wave kernel signature [19]	96.52	98.87
Hough transform [20]	53.99	-
Proposed w/o indexing	93.5	94.6
Proposed with indexing	97.8	96.7

Source: Authors, (2026).

Table 1 presents the performance comparison of the proposed and existing works in terms of precision and recall. On seeing the values obtained, it is clear that the proposed model is more accurate in predicting the accurate feature vectors compared to the other models. The proposed model resulted in better output by accurately finding the text word from the document. The feature extraction step in the proposed work is the crucial step and it helped in improving the overall accuracy. The values for precision and recall are determined based on the accuracy in spotting of the query words from the document. The overall precision of the proposed approach is 97.8% and the recall value is 96.7%. The proposed model is also evaluated without the indexing procedure to prove the significance of indexing in word spotting. Without indexing, the performance of the proposed model is somewhat decreased and the overall precision score became 95.87% and the recall score became 94.13%. Among the compared approaches, wave kernel signature method provided better output. Moreover, the recall value of this method is higher than that of the proposed method. This is because, the document considered in the proposed work is a historic document consisting of several forms of distortions and skews.

IV.3.1 Visualized Analysis Of the Proposed Model

Under this section, visualized results of the proposed word spotting model is presented. The evaluation presents ablation studies by evaluating each module of the proposed work in detail.

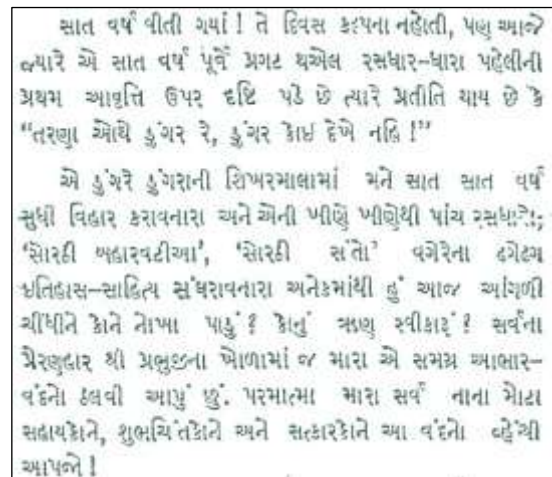


Figure 8: Input text image.

Source: Authors, (2026).

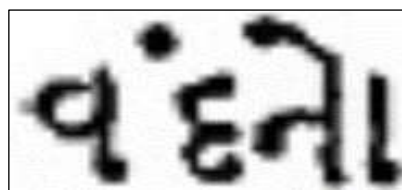


Figure 9: Query image.

Source: Authors, (2026).

Figure 8 presents the input text image considered from the Gujarati text document. It is seen from the input image that there are several noise and distortions that are required to be pre-processed to enhance its quality. Similarly the query image considered in a single experiment is presented in Figure 9. The query image considered consists of 4 characters and the proposed model evaluates the document based on words to determine the accurate word from the document.

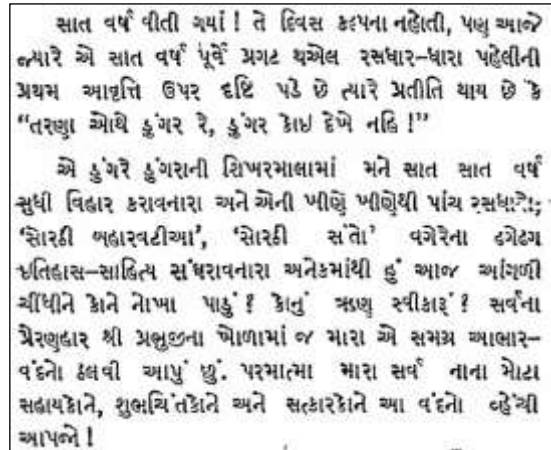


Figure 10: Filtered image.

Source: Authors, (2026).

The initial step in the proposed model is pre-processing and the crucial step in pre-processing is the filtering process. The BF is dedicated for filtering the input image and to enhance the overall quality. The output of the BF technique for the given input text document is displayed in Figure 10. From the figure, it is observed that the BF is capable of enhancing the quality of the input image to a large extent. This filter made the input image suitable for word spotting and then the sequential steps in pre-processing are carried out.

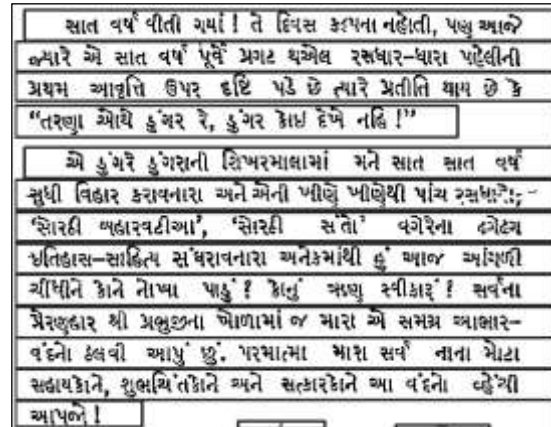


Figure 11: Line segmentation.

Source: Authors, (2026).

After pre-processing the input image, the segmentation process is carried out where the lines are initially segmented from the input image. The output of line segmentation carried out for the considered input image is shown in Figure 11. From the figure, it is seen that the lines are segmented accurately without any overlap between them. The sequential steps followed in line segmentation are executed to obtain the segmented output.

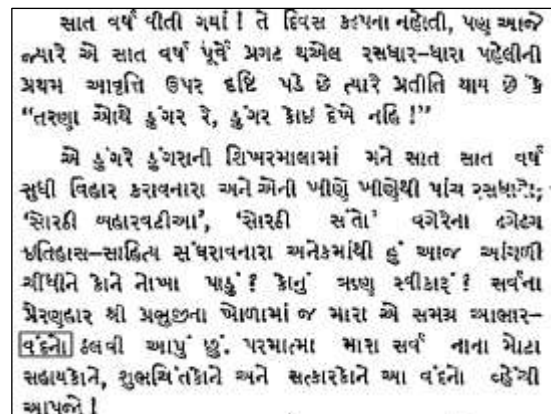


Figure 12: Spotted word in text image.

Source: Authors, (2026).

The final word spotting output is displayed in Figure 12. From the figure, it is clear that the proposed model is capable of accurately locating the word in the complex text document. The main reason for the model to achieve better performance is the feature extraction step. The proposed CAE model extracted the discriminative features from the word images. Thus the extracted features are capable of providing more information about the query image. This step enhanced the indexing and matching processes as there is a vast deviation between the features of different words. The overall simulation results proved that the proposed model is capable of providing accurate spotting results than the other compared models. The extraction capability of the CAE model enhanced the overall performance by providing highly discriminative image features. Due to this step, the discriminations between the features of different words are well noted by the proposed model. Based on the similarities between the features, the ranks are assigned to the images so that accurate output is generated. On comparison, it is also identified that the precision value of the proposed model is much higher than the compared models. This proves that the proposed model is capable of identifying and retrieving the query words accurately from the historic complex text documents.

V. CONCLUSIONS

In this paper, a new and effective word spotting model is proposed to deal with the handwritten documents of Gujarati language. Initially, the input text document is pre-processed to enhance the quality of the image. In the pre-processing stage, the noise and distortions are removed through the use of bilateral filter. Then, binarization and skew detection are carried out to enhance the overall image quality. After this step, line and word segmentation is performed to segment the words separately from the text document. Then, the major feature vectors are extracted from the word images using the 3-layer deep CAE model. The weight values are optimized and selected for the model using the IEHO algorithm to enhance the overall accuracy. After extraction, the extracted features are indexed using the geometric hashing algorithm. Further, matching is done based on four major similarity measures to understand the capability of the model in accurately spotting the words based on query image. The proposed model is implemented and evaluated in python and the results proved that the model is more effective than the compared methods. In future, it is aimed to use different deep neural networks with excellent extraction capability for feature extraction. Also, other than Gujarati script, there are several other scripts that are in need of a perfect word recognition model. Thus, the future scope relies on building models for word recognition for other language documents.

VI. AUTHOR'S CONTRIBUTION

Conceptualization: Bijal Dalwadi, Vatsal Shah.

Methodology: Bijal Dalwadi, Vatsal Shah.

Investigation: Bijal Dalwadi, Vatsal Shah.

Discussion of results: Bijal Dalwadi, Vatsal Shah.

Writing – Original Draft: Bijal Dalwadi, Vatsal Shah.

Writing – Review and Editing: Bijal Dalwadi, Vatsal Shah.

Resources: Bijal Dalwadi, Vatsal Shah.

Supervision: Bijal Dalwadi, Vatsal Shah.

Approval of the final text: Bijal Dalwadi, Vatsal Shah.

VII. REFERENCES

- [1] Benabdelaziz, R., Gaceb, D. and Haddad, M., 2020, May. Word-Spotting approach using transfer deep learning of a CNN network. In 2020 1st international conference on communications, control systems and signal processing (CCSSP) (pp. 219-224). IEEE.
- [2] Sharma AK, Thakkar P, Adhyaru DM, Zaveri TH. Handwritten Gujarati Character Recognition Using Structural Decomposition Technique. Pattern Recognition and Image Analysis. 2019 Apr;29(2):325-38.
- [3] Suthar, S.B. and Thakkar, A.R., 2022. Hybrid Deep Resnet With Inception Model For Optical Character Recognition In Gujarati Language. Reliability: Theory & Applications, 17(1 (67)), pp.194-209.
- [4] Pareek, J., Singhanian, D., Kumari, R.R., Purohit, S. (2020 Jan 1). Gujarati Handwritten Character Recognition from Text Images. Procedia Computer Science. 171: 514-23.
- [5] Dua, Mohit, Bhavesh Bhagat, Shelza Dua, and Nidhi Chakravarty. "A review on Gujarati language based automatic speech recognition (ASR) systems." International Journal of Speech Technology 27, no. 1 (2024): 133-156.
- [6] Ahmed, R., Al-Khatib, W.G. and Mahmoud, S., 2017. A survey on handwritten documents word spotting. International Journal of Multimedia Information Retrieval, 6(1), pp.31-47.
- [7] Dutta, K., Krishnan, P., Mathew, M. and Jawahar, C.V., 2018, August. Towards spotting and recognition of handwritten words in indic scripts. In 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR) (pp. 32-37). IEEE.
- [8] Bhunia, A.K., Roy, P.P., Sain, A. and Pal, U., 2020. Zone-based keyword spotting in Bangla and Devanagari documents. Multimedia Tools and Applications, 79(37), pp.27365-27389.
- [9] Omayio, E.O., Sreedevi, I. and Panda, J., 2021, January. Word Spotting of Handwritten Hindi Scripts by Circular Histogram of Oriented Displacement (CHOD) Features. In 2021 4th Biennial International Conference on Nascent Technologies in Engineering (ICNTE) (pp. 1-6). IEEE.
- [10] Boudraa, O., Michelucci, D. and Hidouci, W.K., 2022. PUNet: Novel and efficient deep neural network architecture for handwritten documents word spotting. Pattern Recognition Letters, 155, pp.19-26.
- [11] Rothacker, Leonard, Fabian Wolf, and Gernot A. Fink. "Annotation-free word spotting with bag-of-features HMMs." International Journal of Pattern Recognition and Artificial Intelligence 35, no. 04 (2021): 2153001.

- [12] Frinken, Volkmar, and Shriphani Palakodety. "Handwritten keyword spotting in historical documents." *Handwritten Historical Document Analysis, Recognition, and Retrieval—State of the Art and Future Trends, Series in MP&AI 89* (2020): 81-99.
- [13] Wolf, Fabian, and Gernot A. Fink. "Annotation-free learning of deep representations for word spotting using synthetic data and self labeling." In *International Workshop on Document Analysis Systems*, pp. 293-308. Springer, Cham, 2020.
- [14] Westphal, Florian, Håkan Grahn, and Niklas Lavesson. "Representative image selection for data efficient word spotting." In *International Workshop on Document Analysis Systems*, pp. 383-397. Springer, Cham, 2020.
- [15] Wolf, Fabian, Kai Brandenbusch, and Gernot A. Fink. "Improving Handwritten Word Synthesis for Annotation-free Word Spotting." In *2020 17th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pp. 61-66. IEEE, 2020.
- [16] Majumder, S., Ghosh, S., Malakar, S., Sarkar, R. and Nasipuri, M., 2021. A voting-based technique for word spotting in handwritten document images. *Multimedia Tools and Applications*, 80(8), pp.12411-12434.
- [17] Cheikhrouhou, A., Kessentini, Y. and Kanoun, S., 2021. Multi-task learning for simultaneous script identification and keyword spotting in document images. *Pattern Recognition*, 113, p.107832.
- [18] Farooqui, F.F., Hassan, M., Younis, M.S. and Siddhu, M.K., 2020. Offline hand written Urdu word spotting using random data generation. *IEEE Access*, 8, pp.131119-131136.
- [19] Das, S. and Mandal, S., 2020. Segmentation-free word spotting in historical Bangla handwritten document using Wave Kernel Signature. *Pattern Analysis and Applications*, 23(2), pp.593-610.
- [20] Kundu, S., Malakar, S., Geem, Z.W., Moon, Y.Y., Singh, P.K. and Sarkar, R., 2021. Hough transform-based angular features for learning-free handwritten keyword spotting. *Sensors*, 21(14), p.4648.
- [21] Hu, Qiubo, Wenxiang Xu, Xiaobo Liu, Zhihua Cai, and Junjie Cai. "Hyperspectral image classification based on bilateral filter with multispatial domain." *IEEE Geoscience and Remote Sensing Letters* 19 (2021): 1-5.
- [22] Su, Bolan, Shijian Lu, and Chew Lim Tan. "Binarization of historical document images using the local maximum and minimum." In *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems*, pp. 159-166. 2010.
- [23] Gatos, Basilios, Ioannis Pratikakis, and Stavros J. Perantonis. "Hybrid off-line cursive handwriting word recognition." In *18th International Conference on Pattern Recognition (ICPR'06)*, vol. 2, pp. 998-1002. IEEE, 2006.
- [24] Muthusamy, Hariharan, Sindhu Ravindran, Sazali Yaacob, and Kemal Polat. "An improved elephant herding optimization using sine–cosine mechanism and opposition based learning for global optimization problems." *Expert Systems with Applications* 172 (2021): 114607.