International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Assamese Script Identification and Content-Based Image Retrieval using Improved Siamese Neural Network

[1*]Sathiyaviradhan Janarthanan, [2]Sivagnana Raj Shanmuganathan

Submitted: 15/11/2022 **Revised:** 25/12/2022 **Accepted:** 15/01/2023

Abstract

Content based image retrieval (CBIR) is a remarkable system that allows users to effortlessly search and retrieve images from a vast dataset. With a large number of images in the dataset, identifying similar images to our query image can be quite challenging. An effective and efficient approach is needed to utilize information from these image repositories. This system allows users to easily access Assamese Script images from the database by searching for specific content. Assamese is a language that falls under the Indo-Aryan language family and is one of the 22 scheduled languages in India. There is a wide range of Assamese printed documents available in digital format. However, retrieving information from these digital document images is a task of the highest priority. Current studies have shown remarkable precision rates in initial retrieval levels, such as the top 10 and top 20 images. However, these rates drop significantly in subsequent levels, like the top 40, 50, and 70. Therefore, the objective of this paper is to introduce a novel CBIR approach that attains high precision values across all retrieval levels. The system consists of three main stages: Preprocessing, feature extraction, and feature similarity matching. Effective feature extraction techniques are crucial for optimizing the performance of a CBIR system. Therefore, we suggest utilizing the Improved Siamese Neural Network (ISNN) as a technique for extracting shape-based features. The Cosine distance with ISNN is utilized to match the query image features provided by users and identify a particular word in the document. The experimental findings demonstrate that, in comparison to the other current approaches taken into consideration in this study, the suggested methods yield superior outcomes.

Keywords: *Assamese script, Grayscale Conversion, threshold technique, CBIR System, Inception V3, Segmentation, word recognition, shape features, deep learning, feature extraction.*

1. Introduction

Advancements in technology have enabled efficient production, processing, storage, and transmission of documents. To progress towards paperless official transactions, numerous printed materials are currently being scanned, digitized, and stored as images in databases. Recently, there has been a significant increase in interest in document image retrieval systems [1,2]. The growing amount of multimedia data necessitates the development of advanced methods for information retrieval. Content-Based Image Retrieval (CBIR) uses image-specific data, mostly through comparing aspects like color, texture, shape, and layout to get comparable images. Recognizing words in a document image is crucial for retrieval [3,4].

Document image retrieval systems (DIRS) are now accessible for printed Roman, Korean, English, Chinese, and other oriental scripts. However, the inclusion of Indian scripts is still rare. This work focuses on creating a document image retrieval system that is independent of font and dimensions for various printed Assamese texts. The Assamese script, often referred to as Asamiya, is an abugida utilized for writing the Assamese language, mostly spoken in the state of Assam in northeastern India

[5,6]. The script belongs to the Brahmic family and has parallels with scripts used in the Indian subcontinent. The script has distinct alphabets, such as vowels (swarabarnas), consonants (byanjanbarnas), and other diacritic symbols used to represent vowel sounds when paired with consonants. The Assamese script contains a diverse range of characters, including a variety of consonants and vowels that display complex variations. The intricacy can provide difficulties for character recognition systems, particularly when facing different fonts and styles of handwriting [7,8].

Feature extraction is the initial step in CBIR that transforms human vision into a numerical representation that can be processed by machines. The quality of the recovered images depends significantly on the extracted features. This selection is reliant on the needs of the user. Utilizing extracted characteristics in machine learning techniques can enhance the efficiency of CBIR. Recent image retrieval research trends focus on utilizing deep learning to enhance accuracy while also improving computational time. High-dimensional characteristics generated during the translation of visual picture content to numerical form might negatively impact CBIR performance in terms of memory utilization, scalability, speed, and accuracy. The high-dimensional feature representations, often known as the "curse of dimensionality," typically exhibit a sparsely dispersed nature. This issue may be resolved through "dimensionality reduction" [9,10].

Several detailed studies in the literature have examined various proposed techniques for dimensionality reduction [11,12]. Similarity measure is a crucial factor that influences the performance of CBIR. Choosing an unsuitable measurement for the feature vector layout can reduce the reliability of the CBIR system by returning fewer identical images. By utilizing an appropriate similarity measure, it is possible to attain high accuracy. Various datasets are utilized in CBIR frameworks, with metrics including precision, recall, and running time commonly employed to assess CBIR performance, which is impacted by the choice of picture dataset. CBIR applications have been utilized in a variety of fields including surveillance systems [13], Geographical Information Systems [14], Remote Sensing [15], Architectural Design [16], Medical Image Retrieval [17], and Object Recognition [18].

Moreover, current CBIR techniques [19-21] utilize different visual attributes such as color, texture, and shape to construct the feature database. The CBIR system outputs the most similar images based on distance, representing the semantic response to the query image. Rahman et al. [22] integrated Euclidean and Bhattacharyya distances to categorize skin lesions using color and texture characteristics. Fuzzy Hamming distance was utilized in [23] to quantify the similarity between the query and dataset images. Naik et al. [24] introduced a boosted distance measure that assigns weights to individual features based on their discriminatory ability in digital histopathology for classifying breast tissues. Ballerini et al. [25] introduced a question by example CBIR system that use weighted Euclidean distance as a similarity metric to categorize skin lesions into five classifications. Visual features-based CBIR systems can improve image retrieval accuracy, but the performance of current CBIR algorithms declines when dealing with many classes. Moreover, the computational expense of these techniques is substantial when dealing with extensive image datasets.

To begin with, the Assamese textdocuments are changed into the image samples. In thewake of preprocessing the noise is expelled and scriptwas segmented utilizing ASRNN model. After that, Inception V3 model is applied to segmented image to extractrelevant features from character to form feature vectors. These feature vectors are then utilized by a novel approach named ISNNto recognize the Assamese word image and for the subsequent similarity matching process. The culmination of this process is the ability to present users with relevant results, revealing the occurrences of queried words within the original documents. By addressing the challenges associated with Assamese script recognition and retrieval, this research contributes to the broader field of document image analysis and retrieval, providing insights and methodologies that can be adapted to other scripts and languages. Our approach would add to decrease theerror, so our proposed strategy achieves high accuracyand furthermore, this approach accurately recognizes thesubstantial volume of words.

The remainder of the paper is structured as follows: Section 2 examines previous research in the topic. Section 3 delves into the suggested method, including its mathematical representation and development, as well as revealing the techniques used for learning and image retrieval. Section 4 carries out the experiments and explains their findings, while Section 5 wraps up the paper and discusses future research directions.

2. Literature Survey

The emergence of digital document archiving and retrieval has required the creation of advanced methods for handling and navigating through extensive text collections. CBIR systems are now a crucial tool for organizing and retrieving information from image-based documents.

Rasheed et al. [26] conducted a study that utilized a pre-trained Convolutional Neural Network (CNN) to identify handwritten Urdu characters, an area with limited existing research. The study utilizes a groundbreaking dataset from 2020. The scientists used an unsupervised technique called autoencoder and CNN for recognizing Urdu handwritten characters. AlexNet is well-known for its success in the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) and is considered a key component of their strategy because of its simplicity and distinctive characteristics, including a greater quantity of filters per layer, the presence of pooling layers alongside stacked convolutional layers, quicker computational speed, and decreased reliance on hardware. The authors present two frameworks for classifying hand-written letters and digits using pre-trained AlexNet neural networks in this study.

Banerjee et al. [27] introduced a method for comparing query word profiles, including both upper and lower cases, with target words' profiles. The query and target words are first binarized, resulting in the creation of two profiles for each word. The authors then developed guidelines to exclude unimportant words in relation to the query word and identify possible query word candidates. The candidate query words' profiles are compared with the query word profiles in the Z-transform domain using the resonance condition for the damped oscillator. An affine transformation is used to adjust the Bezier curve representation of the candidate query words' profiles before matching to address variation writing styles including scaling, rotation, and shearing.

Ghazal et al. [28] developed a system to create a dataset of English language handwritten character images. The system shows promising recognition results in several trials after being extensively

trained on a huge sample data set and tested on user-defined handwritten document images. The system's workflow involves multiple crucial stages, starting with picture pre-processing to enhance data for training using a convolutional neural network (CNN). After the initial processing, the input document is segmented at the line, word, and character levels to enable more accurate analysis and categorization. This thorough method for dataset preparation and segmentation establishes a strong basis for precise character identification in handwritten English documents, representing a notable progress in this study field.

Lakshmi et al. [29] introduced the Telugu Word Image Retrieval (TWIR) system, which utilizes a sophisticated artificial intelligence-based multi-layer deep learning convolutional neural network (DL-CNN) to extract precise properties of Telugu words. AlexNet was utilized to cluster features using a deep learning model known as DeepCluster. By utilizing the DeepCluster approach, numerous grammatical rules of Telugu scripts will be thoroughly evaluated. The feature database was trained using Telugu word characteristics and grammar standards. The system will efficiently retrieve Telugu words and be beneficial in real-time applications like English to Telugu translations, Telugu online browsing, and Telugu speech analysis.

Zagoris et al. [30] presented a method for accurate word spotting in handwritten texts. The method relies on local features that focus on information at crucial areas in the document. Their approach includes a matching mechanism that combines spatial context in a local proximity search, without requiring any training data. Their methodology utilizes document-oriented keypoint and feature extraction, along with a fast feature matching mechanism, to create a simplified pipeline suitable for efficient deployment in the cloud.

After reviewing all of the related literature, it is evident that significant research has been conducted on the recognition of printed and handwritten text in various scripts. Text recognition system for ancient documents is unavailable. The accuracy of script recognition models is frequently influenced by the quality of the input images. Challenges like low resolution, noise, blurring, or distortion may significantly impact performance. This research proposed a deep learning model for Assamese character recognition to address the identified drawbacks.

3. Methodology

The introductory section of this paper discusses the development of the suggested approach, encompassing its mathematical formulation and learning methodology. Following that, the technique for matching the images and retrieving images that are similar to a user query image is described.

The methodology utilized in our study is crucial for the development of a CBIR system that is efficient and specifically designed for the Assamese script. This section outlines the methodical strategy and sequential computational procedures that were implemented in order to overcome the obstacles related to the identification and retrieval of Assamese script from PDF files. The procedure has been carefully crafted to transform the intrinsic visual data of the script into a format that is both searchable and retrievable, enabling users to efficiently conduct queries.

Our methodology is divided into two distinct phases, as depicted in the accompanying figure 1: the training phase and the query phase. Preparing the CBIR system through the development of a

robust dataset and the training of neural network models to identify and extract features from Assamese script constitutes the training phase. The process entails the conversion of PDF documents into single-page images, the application of an advanced Attention segmental recurrent neural network (ASRNN) to segment words, and the utilization of the Inception v3 model to map the segmented images to their respective text files. Following this, an ISNN is employed to extract unique shape-based characteristics from the text files and word images, which are crucial for the retrieval procedure.

On the other hand, the query phase serves as an environment for the practical implementation of the system. The system analyzes unidentified query word images provided by users through the process of mapping them to their textual forms and extracting significant features. A similarity matching algorithm is then applied to these features in relation to the dataset, resulting in the retrieval of the most pertinent outcomes. The system eventually exhibits the page from the document containing the searched word, thereby furnishing users with the intended information in a manner that is easily understandable.

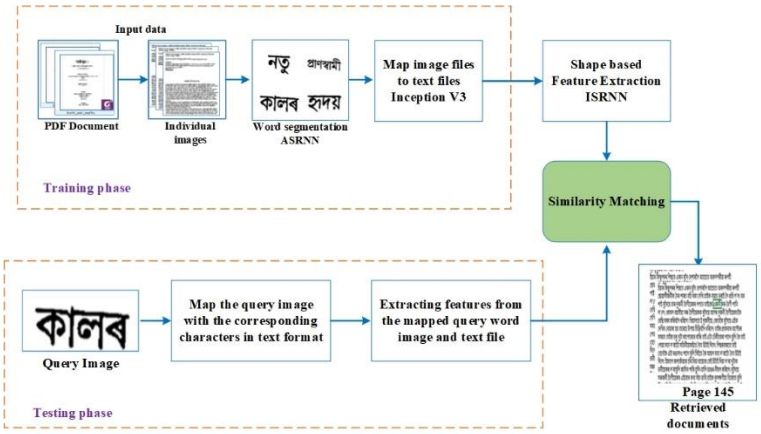


Figure 1. Block diagram of the proposed model

3.1 Dataset description

The provided dataset functions as the fundamental basis for the training stage of a CBIR system. The methodology employed to curate the datasets for the experiments involved a meticulous manual process. The visual depictions were generated by employing publicly available online resources that pertain to the Jonaki and Shankari eras of Assamese literature. The dataset can be made available upon the submission of a formal request, with due consideration given to its potential significance. The dataset utilized in this study was selected and organized with great attention. The compilation comprised images acquired from online publications that were accessible to the public and pertained to the Jonaki and Shankari eras in Assamese literature. The significance of examining these literary epochs lies in their contribution to Assam's cultural heritage. The compilation prominently showcases illustrations from the following five novels: "BezbaruahrRasanawali (Vol 2)," "Kirttana and Ghosha," "Gauburha," and "Jilikoni".

3.2 Preprocessing

The preprocessing phase holds significant importance within the proposed CBIR system, as it is specifically designed to streamline the process of identifying and extracting Assamese script from PDF files. The preparatory pipeline commences

with the transformation of the source PDF files into image formats suitable for single pages.

In order to address the intricacies associated with processing textual data contained within PDFs, the system implements a method whereby every page of the PDF is converted into a separate image. The foundation of this strategy lies in the assumption that image-based processing enables the utilization of a collection of computer vision methods that are not easily implementable on the text and vector graphics that are commonly contained within PDF file formats. With great attention to detail, the conversion procedure is carried out in a manner that guarantees the resulting images maintain the exact appearance of the original script on the PDF pages. Preserving the intricate details of the Assamese characters through the use of high-resolution rendering is critical for subsequent segmentation and feature extraction tasks to guarantee accuracy.

There are two main justifications for this conversion. To begin with, it conforms to established norms in the domain of computer vision, wherein image data serves as the customary input for a multitude of analytical models, especially those that incorporate neural networks. Additionally, it streamlines the process of script recognition by converting the document into a consistent structure that is more readily comprehensible through automated analysis. As a result of this conversion process, an extensive collection of single-page images is obtained, with each image representing an individual page of the initial PDF documents written in Assamese script.

Subsequently, an adaptive thresholding [31] method is implemented, with a noise input image. Adaptive thresholding, which is determined by the image's intensity, is implemented on the input image. The grayscale image comprises the intensity of individual pixels. In this process, the RGB image is transformed into the YCbCr color space, which retains the luminance of black and white pixels (Y), an 8-bit depth for grayscale images, and a 24-bit depth for RGB images. Adaptive thresholding is implemented subsequent to the grayscale conversion of the image (Algorithm 1).

When an input image is provided in grayscale or RGB format, it is converted to a binary image (a white and black image) prior to processing. The primary components of a text image are the (1) text color and (2) background color. The conversion of images to binary format enhances contrast and facilitates the application of Otsu's method for global thresholding [32]. Employing an adaptive local thresholding algorithm, the image's foreground and background are distinguished by their varying intensities. After applying an adaptive (mean/median) filter to emphasize image features, a binary image is produced using the Otsu threshold. On textual data, Otsu's thresholding has been observed to yield favorable outcomes. In order to achieve more optimal preprocessing outcomes, the binarization technique is implemented. Pixels with intensities exceeding a specified threshold are transformed into white; otherwise, they are converted to black. Consequently, the image undergoes a conversion into a combination of black and white pixels. The equation is given below:

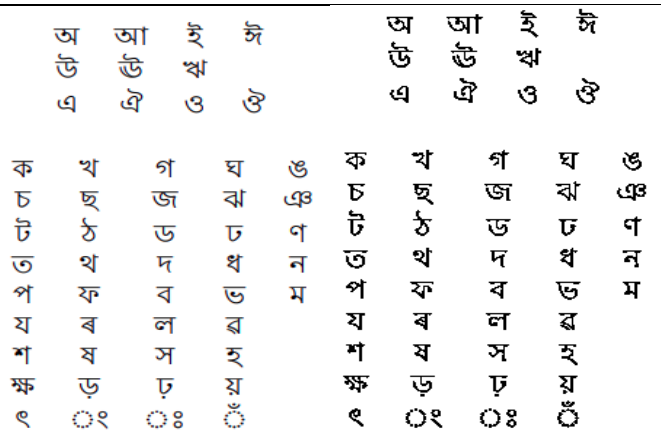
$$\sigma_T^2 = \sum_{i=0}^{L-1} (i - \mu T)^2 Pi \qquad (1)$$

The adaptive threshold algorithm uses an intelligent edge detector to generate an edge boundary for each character, eliminating dot noise through a low-pass filter. When multiple local possibilities exist, a Sobel filter and nonmaximal suppression are used to select the finest pixel for edges.The pseudocode of Adaptive threshold algorithm is

presented in Table1 and the preprocessing result is shown in figure 2.

Table 1. Steps for Adaptive threshold algorithm

Algorithm 1: Preprocessing
Input: Grayscale image or RGB. **Output:** Clean image. //Begin Step 1. //Convert the image from RGB to YCbCr in the first step. YCbCr ← RGB Step 2. //Proceed to eliminate CBCR. The image subsequently goes monochromatic. Step 3. //Implement adaptive thresholding. Step 4. Employ Otsu's method to implement the global image threshold. Step 5. Adaptive (mean → median) filtering to emphasize image features is the fifth step. Step 6. Apply the Otsu threshold to segment and produce a binary image in Step 6. //End



(a) Input image (b) Preprocessed image

Figure 2. Preprocessing Results using Adaptive Thresholding technique

3.3 Word Segmentation Using ASRNN

The segmentation procedure initiates by extracting visual characteristics from the images on a single page. The aforementioned characteristics exemplify the character-level data that is intrinsic to the Assamese script. The words are subsequently dynamically segmented from the images using ASRNN [33]. The network exhibits the ability to effectively process Assamese words of varying lengths and focuses on precise segments of the script in order to isolate individual words. Formulated as a recursive computation, the segmentation procedure determines word boundaries within the script. The attention mechanism integrated into the neural network enables the model to assign varying weights to distinct image components, consequently augmenting the segmentation accuracy. The word-level encoder, which is proficient in managing the complexities of the Assamese script, is represented by this model.

The ASRNN model commences by extracting character-level representations from the single-page image words that have been segmented. These images are obtained from the original PDF documents. Following this, pre-trained GloVe word

embeddings are concatenated with these representations to generate a comprehensive, contextualized feature set. The application of a bidirectional Long Short-Term Memory (Bi-LSTM) network at each timestep k produces a context vector ck. By employing an attentiveness mechanism, an additional Bi-LSTM is recursively utilized to dynamically compute the segmental representation segi for every word segment i. For precise segmentation, this mechanism enables the model to concentrate on the most pertinent portions of the word. The computation of segmental representations occurs for segments with lengths between 1 and L, with L denoting the utmost segment length.

Next, the labels for each segmental representation segi are classified using a fully connected layer. As neural feature scores, the label scores acquired from the fully connected layer are utilized. It is worth noting that a softmax operation is not necessary at this stage for label classification, given that the sum of the scores of the neural features may surpass 1. After integrating the neural feature scores, a semi-Conditional Random Field (semi-CRF) model is constructed. In order to jointly train the semi-CRF model, the scores of the neural features are combined with the conventional semi-CRF features. The integration improves the model's capability to segment words in the image with greater accuracy. The computation methodology employed in the attention-based encoder model is outlined below:

$$c_{it} = \left[\text{forward}_{L^1_{STM}}(x) : \text{backward}_{L_{STM}}(x)\right], \quad (2)$$

$$h_{it} = \left[\text{forward}_{LSTM_2}(c_{it}) : \text{backward}_{LSTM_2}(c_{it})\right], \quad (3)$$

and

$$s_i = \Sigma_t \alpha_{it} h_{it}, \text{ and } F_i = \text{MLP}(s_i) \quad (4)$$

The conditional probability of a potential output sequence s with respect to the input sequence b is represented by the semi-CRF layer as follows::

$$p(s \mid x) = \frac{1}{Z(x)} \exp\{w_1 A(x,s) + w_2 F(x,s)\} \quad (5)$$

Our neural semi-CRF was trained using maximum conditional likelihood estimation. For $\{(x_i, s_i)\}$ training set, log-likelihood is represented as

$$L_D(W) = \Sigma_{i \in D} \log\, p(s \mid x) \quad (6)$$

Despite its computational complexity, the attention-based segmentation strategy prioritizes Assamese script recognition performance increases over standard RNNs. Our attention-based segmentation model takes $O(N_2)$ time, where N is the script image's character length.

3.4 Map image features to binary vectors

After the process of word segmentation is complete, the CBIR system advances to the phase of binary encoding. The segmented word images are transferred to their respective text files utilizing the Inception v3 model during this phase. The purpose of the mapping phase is to build a connection between the textual data (text files) and

the visual data (segmented word images). The objective is to convert the illustrative form of words into a textual format suitable for the query phase processing by the CBIR system. The Inception v3 model is employed due to its capability of identifying intricate patterns present in images. The objective of this context is for the model to discern the characters that are contained within the segmented word images. Due to its extensive training on a wide variety of images, the Inception v3 model has acquired the capacity to precisely identify and categorize the characters comprising the Assamese script.

Szegedy et al. [34] introduced the Inception model, a deep CNN architecture, during the 2014 Large-Scale ImageNet Visual Identification Challenge. The primary objective of this model was to mitigate the impact of computational efficiency and limited parameters in real-world application scenarios. 299 × 299 was the image dimensions entered into Inception-v3 [35]. Despite having a 78% larger footprint than VGGNet (244 × 244), Inception-v3 exhibits a quicker execution speed [36]. The primary factors contributing to the exceptional efficacy of Inception-v3 are outlined below: When juxtaposed with AlexNet [37], Inception-v3 exhibits a parameter count of less than one-fourth that of VGGNet (140,000,000) and less than half that of AlexNet (60,000,000). Moreover, the aggregate number of floating-point computations performed by the Inception-v3 network is an estimated 5,000,000,000, a figure significantly surpassing that of Inception-v1 (approximately 1,500,000,000). The practicality of Inception-v3 is enhanced by these attributes; it can be readily deployed on a shared server in order to deliver a rapid response service.

By utilizing convolutional kernels of varying diameters, Inception-v3 acquires receptive fields encompassing distinct regions. In order to decrease the network's design space, a modular system is implemented, which is subsequently completed by joining, thus enabling the integration of features at different dimensions. A summary of the Inception-v3 network parameters is provided in Table 2.

A batch normalization (BN) layer is incorporated into Inception-v3 to serve as a regularizer in the intermediate layer between the fully connected (FC) and the auxiliary classifier. The sequential gradient descent method can be utilized in the BN model to expedite the deep neural network's training and convergence. The BN formulas are denoted as follows:

$$\mathrm{B} = \{X_{1\ldots m}\}, \gamma, \beta \tag{7}$$

$$\{y_i = BN_{\gamma,\beta}(\chi_i)\} \tag{8}$$

$$\mu_{\mathrm{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} X_i \tag{9}$$

$$\sigma_{\mathrm{B}}^{\ 2} \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathrm{B}})^2 \tag{10}$$

$$\hat{X}_i \leftarrow \frac{X_i - \mu_{\mathrm{B}}}{\sqrt{\sigma_{\mathrm{B}}^2 + \varepsilon}} \tag{11}$$

$$y_i \leftarrow \gamma\hat{X}_i + \beta = BN_{\gamma,\beta}(\chi_i) \tag{12}$$

Table 2. Network structure of the Inception-v3 model

Type	Patch size/stride	Input size
conv	3 × 3/2	299 × 299 × 3
conv	3 × 3/1	149 × 149 × 32
conv	3 × 3/1	147 × 147 × 32
pool	3 × 3/2	147 × 147 × 64
conv	3 × 3/1	73 × 73 × 64
conv	3 × 3/2	71 × 71 × 80
conv	3 × 3/1	35 × 35 × 192
3 × Inception	————	35 × 35 × 288
5 × Inception	————	17 × 17 × 768
2 × Inception	————	8 × 8 × 1280
pool	8 × 8	8 × 8 × 2048
linear	logits	1 × 1 × 2048
softmax	classifier	1 × 1 × 1000

Moreover, large convolution kernels are subdivided into small convolution kernels in series within Inception-v3. Convolution and pooling are linked in parallel, and LSR labels are incorporated to facilitate regularization in accordance with the smoothing criteria. BN is also incorporated into Inception-v3 in consideration of the distribution inconsistency between inputs and outputs in conventional DNNs, which poses significant challenges for feature extraction. By normalizing the input into each layer, it is possible to optimize the learning effect. Modify the last few layers of the Inception v3 model in order to generate a binary vector that symbolizes the encoded characteristics of the input image. An assortment of Assamese script examples are presented in Table 3:

Table 3. Binary vector Encoding

বেজবৰুৱা: [1, 1, 1], [1, 1, 0], [1, 1, 1], [1, 1, 1], [1, 0, 1], [0, 0, 1], [1, 0, 1] ব (ba): Horizontal stroke, vertical stroke, curved lines এ (e): Horizontal stroke, curved lines জ (ja): Horizontal stroke, vertical stroke, curved lines ব (ba): Horizontal stroke, vertical stroke, curved lines ৰ (ra): Horizontal stroke, curved lines ু (u): Curved lines ৱ (wa): Horizontal stroke, curved lines
স্তম্ভিত: [1, 0, 1], [0, 0, 0], [1, 1, 1], [1, 0, 1], [0, 0, 0], [1, 1, 1], [0, 0, 1], [1, 1, 1] স (sa): Horizontal stroke, curved lines ্ (virama): No significant features ত (ta): Horizontal stroke, vertical stroke, curved lines

ম (ma): Horizontal stroke, curved lines ◌় (virama): No significant features ভ (bha): Horizontal stroke, vertical stroke, curved lines ি (i): Curved lines ত (ta): Horizontal stroke, vertical stroke, curved lines
নাচোন : [1, 1, 1], [0, 0, 1], [1, 1, 1], [0, 0, 1], [1, 1, 1] ন (na): Horizontal stroke, vertical stroke, curved lines া (a): Curved lines চ (cha): Horizontal stroke, vertical stroke, curved lines ো (o): Curved lines ন (na): Horizontal stroke, vertical stroke, curved lines

3.5 Feature Extraction

Features are distinctive and suitable characteristics that define an image. Especially crucial when the size of the image data is excessive for direct processing. Due to the varying dimensions and orientations of database images, feature extraction is an essential system task that must be performed in a manner that is distinct for each image. The process by which the input image is transformed into a set of features is referred to as feature extraction [38]. Numerous image representation features have been suggested in the field of pattern recognition. High-level and low-level features predominate, representing the user and image perspectives, respectively. Common low-level features include color, shape, texture, and others. Using an ISNN, we extract shape-based features from the mapped text files and images.

Simple geometric characteristics can classify shapes with substantial variations. The following characteristics are computed on the basis of shape: area, length of the main axis, length of the minor axis, and so forth. The selection of these characteristics is predicated on the knowledge that they will be simple to calculate while simultaneously enabling us to differentiate the inputs. In order to calculate the features, the images were divided into four sections based on their center of mass. By designating the Center of Mass as the segmentation point, an adequate number of object pixels are present in each quadrant, enabling the extraction of accurate features from the images.

- **Area:** The area scalar property quantifies the quantity of pixels within a given region of the entire image.
- **Major Axis Length:** Produces a scalar value representing the ellipse's major axis, which possesses the identical normalized second central moments as the region encircling the object.
- **Minor Axis Length:** It yields a scalar value representing the minor axis of an ellipse whose region encircles the object and possesses the same normalized second central moments.
- **Eccentricity:** The ratio of the distance between the foci of the ellipse to the length of its main axis determines eccentricity, which shares the same second-moments as the region.
- **Orientation:** The angle between the major axis and x axis of the ellipse with the same second-moments as the region is denoted by this scalar value.
- **Convex Area:** It represents the count of pixels within a convex image.
- **Euler Number:** Corresponds to the area of the region divided by the diameter of a circle.

- **Equiv Diameter:** Specifies the percentage of pixels in the region that are also present in the convex hull.
- **Solidity:** The function yields a scalar value that signifies the ratio of pixels in the convex hull that are also contained within the region.
- **Extent:** The scalar value in question is equal to the area of the bounding box when calculated.
- **Perimeter:**The distance between each adjacent pair of pixels along the region's border is denoted by a scalar.
- **Circularity ratio:**The reciprocal of the area of the bounding circle and the area of the shape in question.

3.6 Improved Siamese neural network

Metric learning, or similarity learning, is a method that calculates the distance between two samples to assess their similarity, aiming to minimize separation between similar and dissimilar samples. It is commonly used in small-sample classification, using Euclidean distance and cosine similarity as distance functions.

Given two samples, x_i and x_j, $f(\cdot)$ represents the feature mapping of the samples, and their Euclidean distance in the metric space can be described as Eqn. (13):

$$d_f(x_i, x_j) = d\left(f(x_i) - f(x_j)\right) = \|f(x_i) - f(x_j)\|_2 \tag{13}$$

The main objective of metric learning is to acquire a suitable mapping function while following to specific constraints.

3.6.1 Siamese neural network

SNN [39] is a metric learning technique that addresses the classification problem when dealing with limited samples by quantifying the similarity between two samples. This approach utilizes two weight-sharing subnetworks to process two input samples concurrently, yielding the resultant similarity between the two samples. Through the process of pairing training samples into the model, the training times can be significantly enhanced, allowing for a more thorough exploration of the relationships between different samples. Initially, the model takes the two input samples, $(X_1, X_2)$and maps them to a low-dimensional feature space. It subsequently computes the Euclidean distance between the two resulting feature vectors. Distance is utilized as a metric for evaluating the similarity between samples. Significant disparity signifies a lack of resemblance, while a minimal disparity suggests a strong resemblance. Figure 3 displays the structure of a Siamese network.

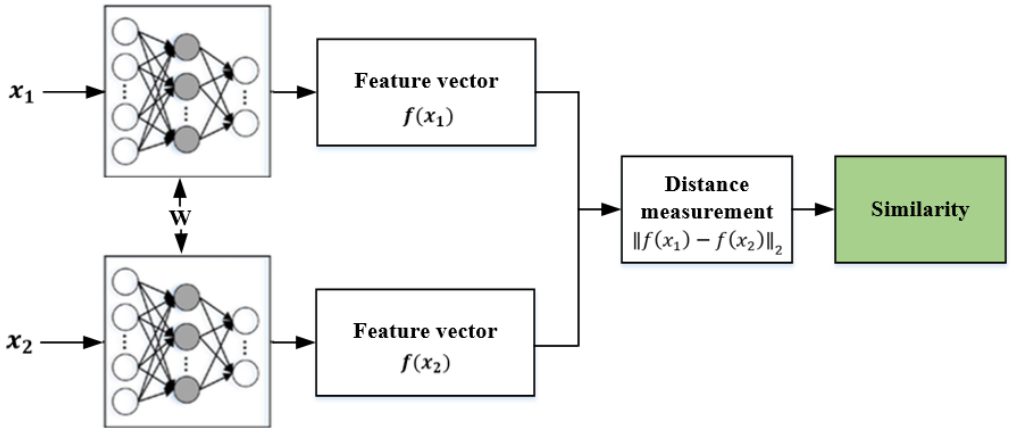


Figure 3. Structure of Siamese network

In our CBIR system, the ISNN is vital to feature extraction, as seen in the workflow diagram. The network rapidly processes pairs of mapped word pictures and text files to obtain shape-based information for similarity matching. When two samples are comparable, the system suggests the word images and text files are from the same Assamese script. This improves document page retrieval accuracy.

The workflow diagram (figure 1) shows our CBIR ISNN structure and ability. The network output shows the query image's similarity to dataset entries as the system processes a query. A high similarity value, near 1, suggests a significant match between the query word picture and a dataset entry, while a low score, around 0, indicates no match. Equation (14) shows how the Siamese neural network optimizes model training target with a contrast loss function:

$$L(\mathbf{w}, (x_1, x_2, y)) = \frac{1}{2} y \|f(x_1) - f(x_2)\|_2^2 + \frac{1}{2}(1-y)\max(m - \|f(x_1) - f(x_2)\|_2, 0)^2 \quad (14)$$

x represents the input sample, y is its similarity label, and y=1 shows similarity between the two samples. If the feature space cosine distance is high, the present model is inefficient and increases loss. y=0 shows no similarity between samples. Loss value increases with tiny cosine distance between samples in feature space. m is the threshold, N is the sample count, and $\|\cdot\|_2$ is the cosine distance between features.

3.6.2 Introduction to ISNN network structure

The traditional Siamese network uses cosine distance for measurement, but its effectiveness depends on the quality of features extracted during initial stages. To maximize each training sample, an efficient network design is needed. The standard SNN has been modified to suit Assamese script recognition needs, allowing for more refined feature extraction and customized similarity measurement. The advanced model integrates a classification branch into the Siamese framework, enhancing the unique characteristics of the Assamese script data.The traditional Siamese network uses cosine distance for measurement, but its effectiveness depends on the quality of features extracted during initial stages. To maximize each training sample, an efficient network design is needed. The standard SNN has been modified to suit Assamese script recognition needs, allowing for more refined feature extraction and customized similarity measurement. The advanced model integrates a classification branch into the Siamese framework, enhancing the unique characteristics of the Assamese script data. As a result, the proposed ISNN for Assamese script recognition includes three essential components: feature extraction, relationship measurement, and script classification. The feature extraction network is responsible for processing the segmented word images and extracting important shape-based features. The relationship measurement component utilizes an advanced Siamese architecture to assess the similarity between features extracted from the query image and those present in the dataset. At last, the script classification network uses the similarity information to accurately find the appropriate instances of Assamese script within the PDF documents.

3.6.3 Feature extraction network

As part of the data preprocessing stage for our CBIR system, we process segmented word images from Assamese script PDF documents to use as input for the feature extraction network. The

feature extraction network is built using an advanced Siamese neural network architecture. It is composed of two parallel subnetworks that have the same structures and share parameters. The submodule initially utilized two LSTM layers (L1 and L2 in Figure 4) to capture the temporal characteristics of the fault sample. Subsequently, the spatial information was extracted using the convolutional layer C1. Lastly, a P1 layer was added to down-sample the convolution results and decrease the feature size.

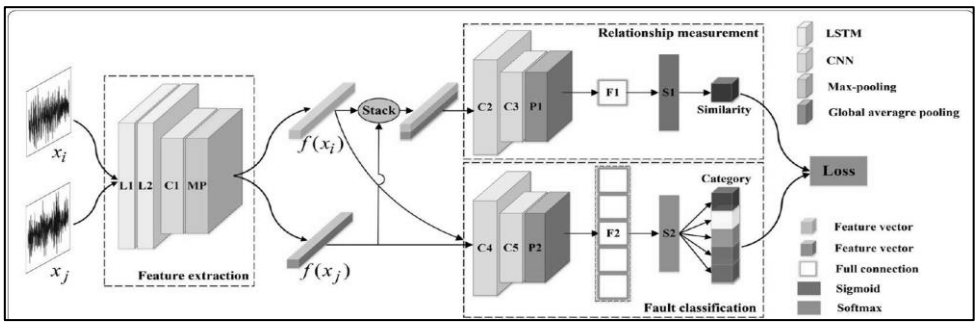


Figure4. Structure diagram of ISNN model [44]

The feature extraction process aims to capture the intricate geometric patterns and contours of the Assamese characters. Following the initial feature extraction, the network utilizes a sequence of convolutional layers to enhance the spatial information. This is then followed by pooling layers to decrease the dimensionality of the feature representation. This method guarantees that the most important aspects of the script are highlighted, getting the data ready for the following phase of measuring relationships.

The ISNN system enhances the training process by incorporating mapped word images and text files, resulting in a broader analysis of the connections between different samples of the Assamese script.

3.6.4 Relationship measurement network

The relationship measurement network in our CBIR system assesses the similarity between pairs of feature vectors extracted from Assamese script images $f(X_i)$ and their corresponding text files (X_j). If two samples are similar, the output probability is 1. Conversely, if two samples are not similar, the output probability is 0.

Many commonly used measurement methods heavily depend on the spatial quality of the feature embedding learned by a feature extraction network. In this study, neural networks were utilized to assess the similarity relationship between features. The networks were trained in conjunction with the feature extraction network, and the measurement method was dynamically adjusted based on the input features. From Figure 4, it is evident that the ISNN model structure includes the relationship measurement network, which consists of convolutional layers C2 and C3, a global average pooling layer P2, and a fully connection layer F1. The feature map was replaced with its average value using global average pooling. Many model parameters were significantly reduced. Initially, the feature vectors $f(X_i)$ and $f(X_j)$ underwent processing in the convolutional layer. Subsequently, they were transformed into a similarity value via the P2 and F1 layers. Finally, the activation function layer S1 was utilized to convert the similarity value into a range of [0,1]. The similarity value was calculated using Equation (15).

$$R_{i,j} = \text{Sigmoid}\left(g\left(f(X_i), f(X_j)\right)\right) \quad (15)$$

The activation function used is the sigmoid function.

A weighted similarity loss function was defined to ensure accurate measurement of the similarity between samples of easily confused categories. This loss function takes into account a penalty coefficient that is determined by the complexity of differentiating between various categories of the Assamese script. As a result, it amplifies the loss for incorrect assessments made among closely related script samples. The loss function is provided in Equation (16).

$$L_S = \sum_{i,j=1,i\neq j}^{N} \left[Y_{i,j}\left(1 - R_{i,j}\right)^2 + \alpha_{i,j}\left(1 - Y_{i,j}\right)\left(0 - R_{i,j}\right)^2\right] \quad (16)$$

where similarity loss is represented by L_S.

3.6.5 Classification network

The relationship measurement network evaluates the resemblance between Assamese script images and text files, but it lacks the ability to categorize or retrieve script instances. To address this, a classification network is incorporated into the ISNN model. The classification network consists of convolutional layers C4 and C5, a global average pooling layer P3, and a fully connected layer F2. An activation function, S2, generates predicted probability for each script. In the training phase, the network uses pairs of segmented word images and text files to determine their categories. In the testing or query phase, only one instance is needed for classification, and the cross-entropy loss function is used to compare the predicted label distribution with the true label distribution of the script samples.

$$L_C = \frac{1}{2N}\sum_{i,j=0}^{N} \left[(Y_i - y_i)^2 + \left(Y_j - y_j\right)^2\right] \quad (17)$$

The formula for calculation can be found in Equation (18).

$$y = \text{Softmax}\left(h(f(X))\right) \quad (18)$$

At the start of our CBIR system, the feature extraction network extracts relevant features from input data. The relationship measurement network then trains the network using similarity information to ensure that script samples with similar features converge and those with different categories diverge. The classification network classifies script instances to retrieve them. Even with a limited sample size, the ISNN model's feature extraction, relationship measurement, and classification interdependently use Assamese script data's visual and linguistic information. Our approach controls parameter size by maintaining a shallow network topology using metric learning for categorization. To train the model, the similarity loss L_S and classification loss L_C were tuned simultaneously and merged to determine the final loss function (Eqn. 19).

$$\text{Loss} = L_S + L_C \quad (19)$$

Our integrated approach ensures the efficient recognition and retrieval of Assamese script in our CBIR system, presenting accurate and relevant search results to users.

3.6.7 Similarity measure

The similarity measure [40] is crucial in conjunction with feature extraction in CBIR. The similarity index calculates the distance between the features of the query image and the features of each image in the database. It uses a distance metric to measure this distance and then ranks the images based on this value. Indexing is done based on the value of this similarity index. The indices are arranged in ascending order to ensure that the image with the closest distance to the query image is displayed at the top of the retrieved images list. The implementation of cosine distance is provided here. The cosine distance can be calculated between a query image and each image in a dataset or stored feature information in a database. The calculation of the cosine distance involves:

$$\text{Similarity} = \cos\,\theta = \frac{\vec{Q}\cdot\vec{D}}{|\vec{Q}|\cdot|\vec{D}|} \quad (20)$$

The measure of similarity between two vectors is determined by the similarity returned from the operation in equation 1. The cosine distance is a numerical value that falls within the range of 0.0 to 1.0. A cosine similarity value of 1.0 indicates that the two vectors are exactly the same.

3.6.8 Retrieval of documents

A comprehensive search utilizing cosine distance is employed to generate a ranking of the top-k candidates for each query (refer to Figure 5). Performance measures for each query included Average Precision (AP) and Recall.

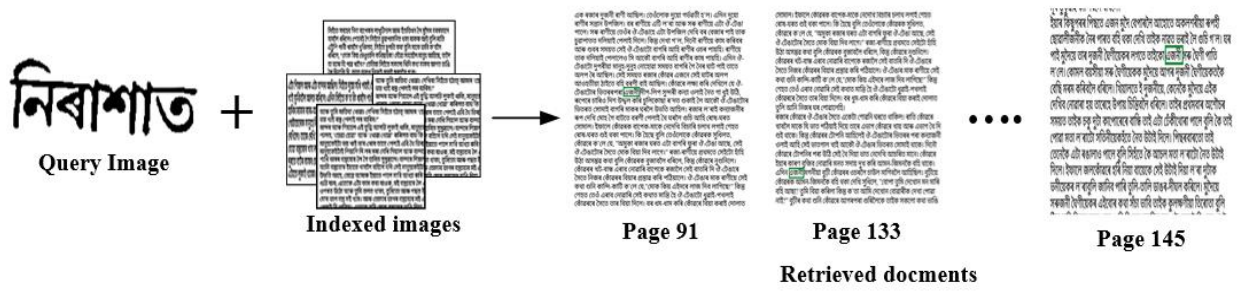


Figure 5. Example of the results shown in the image retrieval task

4. Experimental Results

The complete implementation was done on a Windows 10 system using the Python programming language. The test is written using the Keras framework and utilizes Tensorflow as the backend. An Intel(R) Core (TM) i3 CPU M380 @2.53GHz with 4GB of RAM is being used to analyze the proposed model. The evaluation results of the proposed model differ from those of the present classification, segmentation, and feature extraction strategies. In addition, the proposed method is analyzed using various metrics such as accuracy, precision, recall, F-measure, error rate, and Kappa. These metrics are compared to those of existing algorithms.

4.1 Data Collection

The dataset was created by converting each page of the aforementioned PDF documents into high-resolution single-page images. This conversion process resulted in a total of 2,500 images, ensuring a comprehensive coverage of the script's various stylistic nuances.The selected PDF documents are as follows:"Burhi_Aair_Xadhu", "BezbaruahrRasanawali (Vol 2)", "Kirttana_and_Ghosha", "Gauburha" and "Jilikoni".

We divided the dataset into two subsets, namely training and testing, in order to facilitate the training and evaluation of our CBIR system. The training set consists of 5,000 images, accounting for roughly 80% of the entire dataset. The neural network models were trained using these images, which included the Attention segmental recurrent neural network for word segmentation and the ISNN for feature extraction.

Out of the dataset, the testing set was assigned the remaining 1500 images, resulting in for 20% of the entire set. This subset is crucial for evaluating the system's ability to accurately identify and retrieve Assamese script from confused data. The testing set enables us to assess the precision and recall of the system, as shown in the precision-recall graph, ensuring that the models generalize effectively to new instances of the script. Figure 6 displays the input image retrieved from the database.

Figure 6. Sample images from the present database

We implemented an adaptive thresholding technique to tackle the issue of varying lighting conditions and contrast levels across the document images. This approach allows for the automatic adjustment of the threshold value for each pixel, taking into account the unique characteristics of the local image. This technique proves to be highly effective, especially when dealing with images that have uneven illumination. Using adaptive thresholding, the Assamese script can be effectively binarized, resulting in a clearer differentiation between the text and the background.

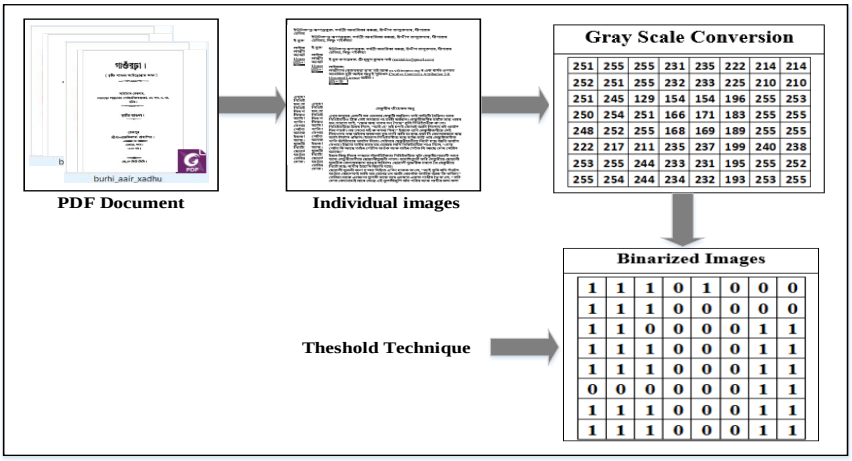


Figure 7. Flowchart of Preprocessing step

Figure 7 shows the preprocessing method used in creating our CBIR system for recognizing Assamese script. The page images are converted to grayscale as a crucial preprocessing step to simplify them by removing color information. Grayscale images are depicted by a matrix of pixel intensity values, varying from 0 for black to 255 for white. This conversion emphasizes the text by reducing the images to a single brightness channel, which enhances the contrast between the text and the background. The photos are processed with an adaptive thresholding technique after being converted to grayscale. This technique utilizes a threshold value that is distinct to each pixel, determined by the local image properties. This technique's adaptability enables effective binarization of the script, especially when dealing with uneven illumination or background textures. The result of the adaptive thresholding procedure is a collection of binary pictures, with the text depicted by pixel values of '1' (white) and the backdrop by '0' (black). Binary pictures are essential for accurately detecting and segmenting the Assamese script because they clearly differentiate the text from its surroundings. Figure 8 displays the outcome of the data preparation.

Figure8. Preprocessing Results

4.2 Evaluation Metrics

The ISNN classifier suggested in this section is evaluated using performance validation metrics for word recognition. The evaluation of the classification results is based on quantitative metrics like as accuracy, precision, recall, kappa, and F-measure.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (21)$$

$$Precision = \frac{TP}{TP+FP} \quad (22)$$

$$Recall = \frac{TP}{TP+FN} \quad (23)$$

$$F1\ score = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (24)$$

$$Kappa = \frac{(observed\ accuracy - expected\ accuracy)}{(1 - expected\ accuracy)} \quad (25)$$

$$mAP = \frac{1}{|T|}\sum_{Q \in T} AP(Q) \quad (26)$$

where T is the set of test images or queries Q.

4.3Training and testing validation

The suggested model was applied to the specified Assamese dataset and assessed based on recognition accuracy. The suggested model in Figure 9 shows a clear trend of increased training-testing efficiency and decreasing training-testing loss as the number of epochs increases. The model has an accuracy of 97.25% on both the training and testing datasets. The outcome was achieved following 600 epochs of model training. The figures below display the training loss and accuracy for both the training and testing sets.

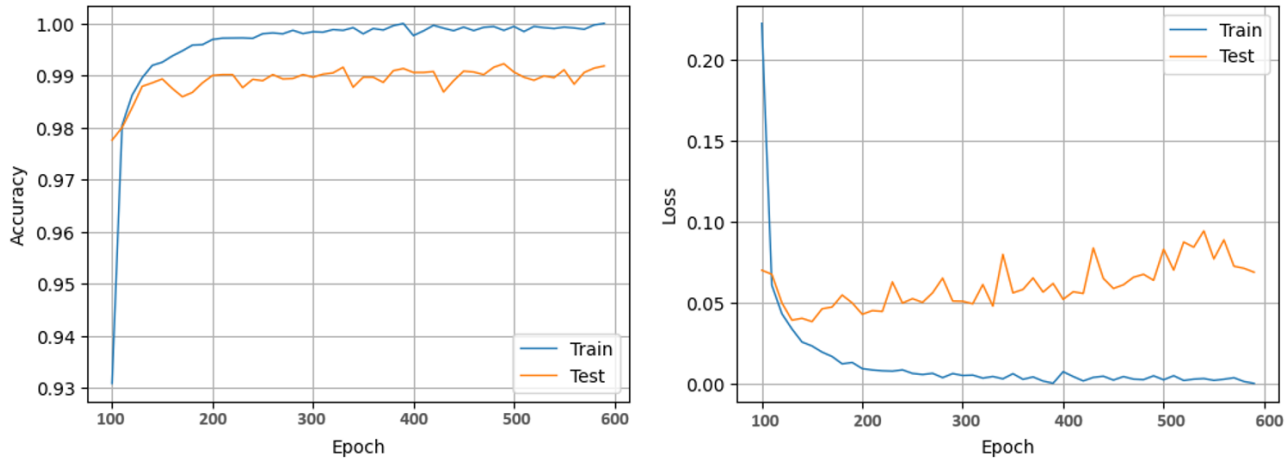


Figure 9. (a) Training and testing accuracy (b) Training and testing loss

There is no overfitting seen in the model. The classification accuracy and loss of the training and validation datasets converge at the 600th epoch. The accuracy of the categorization on the test set is 98.75%. The model has high accuracy in its predictions. The confusion matrix indicates minimal misclassifications in the predictions.

4.4Results and Discussion

The experiment utilized the Adam method to optimize the model parameters with a batch size of 50, a learning rate of 0.002, and a maximum of 600 iterations.

The SNN architecture, trained using the ImageNet dataset, produces 4096-dimensional feature maps that match to its original fc7 layer. We have explored adding a new layer (fcnew) to decrease the feature maps to size of 512, 256, and 128. The values were utilized to decrease the dimensionality of the feature map in a CNN model. Furthermore, we opted to utilize the SNN as a feature extractor to prevent redundant processing of each candidate image. The network feature maps' similarity was computed using the cosine distance. The Top-k candidates were selected to create a ranked list of relevant image candidates based on the mAP, with k values of 25, 50, 75, and 100.

Table 4 displays qualitative outcomes of the images obtained using the ISNN utilizing the complete feature map (4096 dimensions). The results are promising as a significant number of accurate images were found inside the Top-5 ranking. We noticed excellent performance, particularly in the fifth row, where the query closely resembled a signature, and there were no false positives in the Top-5 ranking. We observed some inaccurate candidates in the final row, likely influenced by the

limited number of positive samples in the dataset related to this query.

Table 4. Qualitative retrieval results. Query and the first five retrieved logos using 4096-dimensional feature map

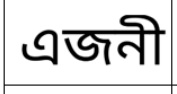
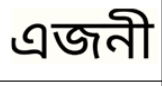
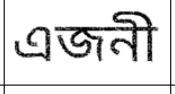
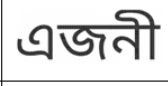
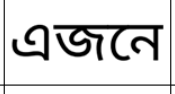
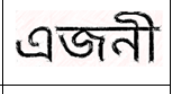
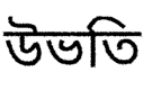
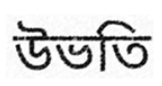
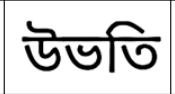
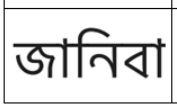
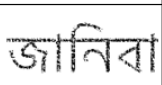
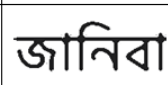
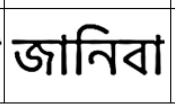
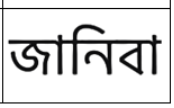
Query Image	Retrieved				
	1st	2nd	3rd	4th	5th
					
					
					

Figure 10 displays the mAP scores, demonstrating the CBIR system's effectiveness in reliably recovering Assamese script from a large dataset. The scores range from 0.91 to 1.00, reflecting the system's precision in finding relevant occurrences of Assamese script from the dataset. A score of 1 indicates flawless precision, with all recovered occurrences being relevant, whilst levels around 0.91 indicate a little reduction in precision. The whole outcome highlights the system's potential as an essential instrument for digitizing and searching Assamese literary texts, aiding in the preservation and accessibility of this linguistic heritage.

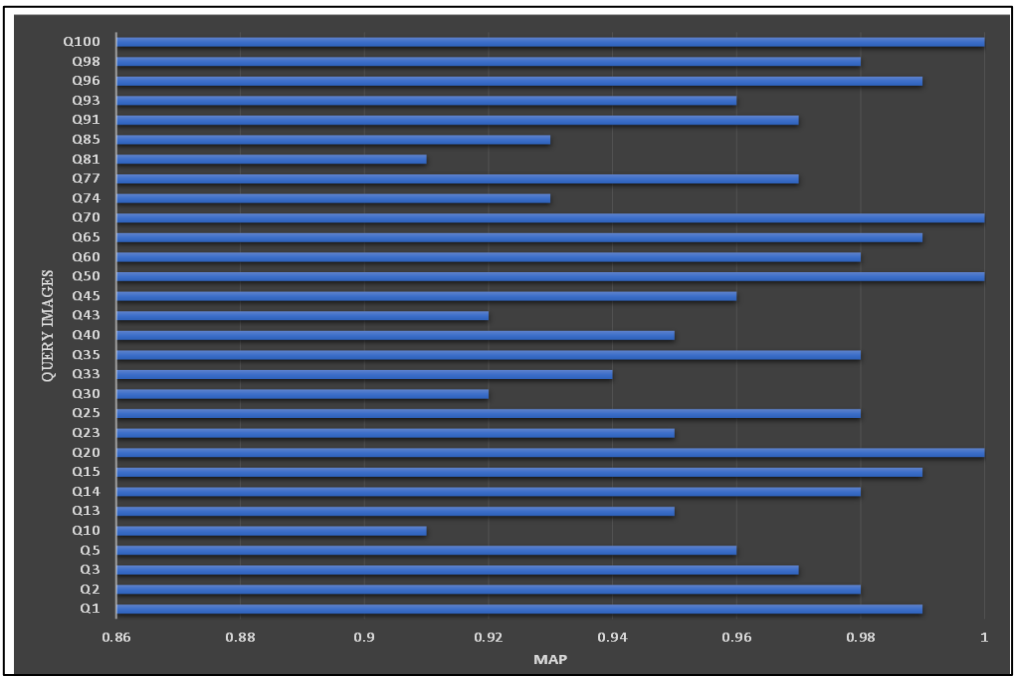


Figure 10. Results by category using ISNN with 4096-dimensional feature map

Table 5 displays the outcomes of a retrieval task using cosine distance as the similarity measure, emphasizing the accuracy rates at the top-50 and top-100 levels for different dimensional feature maps in our suggested CBIR system for Assamese script identification. The feature maps assessed in

the retrieval task consist of 4096-dimensional, 512-dimensional, 256-dimensional, and 128-dimensional representations. The different dimensions represent the level of detail and the quantity of information contained in the feature vectors generated from the Assamese script images. The top-k accuracy metric evaluates the system's capability to identify the accurate script instance among the top 'k' outcomes. In this instance, 'k' is assigned the values of 50 and 100, corresponding to the top-50 and top-100 retrieval possibilities, respectively. The mean Average Precision (mAP) scores range from 0.972 for the 4096-dimensional feature map to 0.655 for the 128-dimensional feature map in the top-50 accuracy evaluation. The top-100 accuracy ratings vary from 0.935 for the 4096-dimensional feature map to 0.506 for the 128-dimensional feature map. Higher-dimensional feature maps result in improved retrieval performance, with the 4096-dimensional feature map attaining the greatest mAP scores in both top-50 and top-100 scenarios. A more comprehensive feature representation that can capture intricate details of the Assamese script leads to improved accuracy and dependability in retrieval. On the other hand, when the number of features in the map drops, there is a clear fall in retrieval accuracy. Lower-dimensional feature maps, although computationally economical, could not have enough detail to successfully differentiate between similar script instances.

Table5.Result of retrieval task using Cosine distance and Top-50 and Top-100

Proposed	$\boldsymbol{Top-k}$	
	50	100
4096-dimensional feature map	0.972	0.935
512-dimensional feature map	0.640	0.486
256-dimensional feature map	0.654	0.503
128-dimensional feature map	0.655	0.506

Table 6 presents a brief overview of the retrieval task outcomes based on cosine distance as the similarity measure. The system scores an accuracy score of 0.972 for the top-50 results and 0.935 for the top-100 results on the 4096-dimensional feature map, demonstrating a high level of precision in finding the most relevant Assamese script instances. Decreasing the dimensionality of the feature map resulted in reduced accuracy. Specifically, the 128-dimensional feature map produced scores of 0.655 for the top-50 results and 0.506 for the top-100 results. The results indicate that feature maps with higher dimensions capture more specific information about the Assamese script, leading to improved retrieval performance. One must evaluate the balance between dimensionality and computational efficiency, as larger feature maps demand increased storage and processing capabilities.

Table 6. Comparison with the state-of-the-art retrieval task using cosine distance and Top- *k* ranking for 4096, 512, 256 or 128 -dimensional feature map.

Author	Feature Map	Top- k			
		25	50	75	100
Proposed Method	4096	0.99	0.96	0.97	0.98
	512	0.64	0.48	0.35	0.22
	256	0.65	0.50	0.36	0.22
	128	0.65	0.50	0.37	0.22

Table 7 illustrates the computational trade-offs related to the dimensionality of feature maps in the CBIR system. The duration of feature extraction is documented for each feature map dimension. The extraction time for the 4096-dimensional feature map is 40.27 seconds, representing the time used to process and extract features from the input data. As the number of dimensions reduces, the time required for feature extraction increases. Specifically, the 256-dimensional feature map took 62.17 seconds, while the 128-dimensional feature map took 54.42 seconds. The retrieval time, given in seconds, is the timeframe needed to complete the retrieval task with the extracted characteristics. The 4096-dimensional feature map has a retrieval time of 35.12 seconds, whereas the 128-dimensional feature map takes 85.14 seconds for retrieval. Higher-dimensional feature maps can expedite feature extraction and retrieval but may necessitate increased processing resources.

Table7. Computational time for the retrieval task (in number of candidates processed per second)

Siamese Model (feature map dimension)	Feature Extraction (sec)	Retrieval (sec)
4096	40.27	35.12
512	57.46	77.45
256	62.17	84.12
128	54.42	85.14

4.5Experimental results on Printed Document Images

This section displays experimental findings of word segmentation conducted on compressed machine-printed document pictures. Segmentation is an essential preprocessing stage in text recognition that aims to isolate words and characters from each other and from the background. This technique allows for more precise recognition and analysis. Figures 11 and 12 display the segmentation outcomes of a CBIR system implemented using Assamese script. The segmentation process seems to have been successfully carried out, as indicated by the distinct separation of words in the text block. The green bounding boxes around each word indicate that the system has accurately recognized and separated the words from the rest of the text.

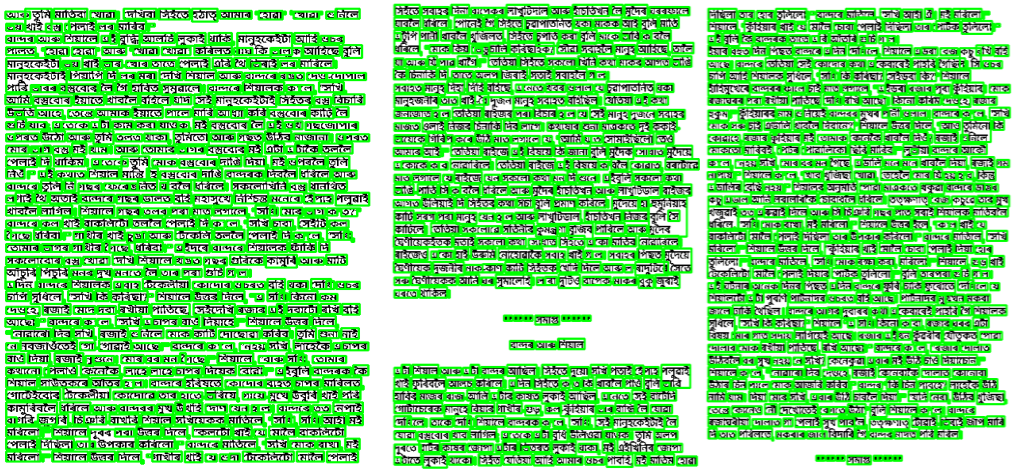


Figure 11.Segmentation result

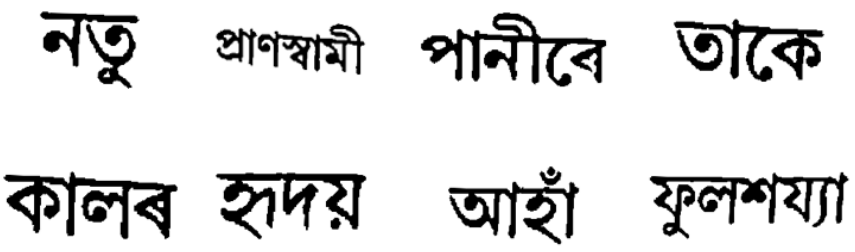


Figure 12. Segmented word image

4.6 Comparative Analysis

The table displays major performance metrics: Accuracy, Precision, Recall, F1 Score, and Kappa statistic corresponding to values 8, 9, 10, 11, and 12. The models examined in this study are the proposed model, NFC-Net, CNN, ANN, DNN-HMM, AlexNet, and LSTM-CNN. Each model is evaluated on its performance using a specific set of query photos labeled Q1 through Q100.

Table 8 and figure 13 display the accuracy analysis comparing the suggested method with existing methodologies. High accuracy reflects a model's proficiency in identifying and retrieving the accurate script, crucial for applications requiring precise script recognition. The proposed model consistently demonstrates high accuracy for all query images, achieving scores between 0.92 and 1.00. The model's great effectiveness in recognizing Assamese script indicates that it may include particular features or procedures designed specifically for the script's unique attributes. NFC-Net and CNN models exhibit strong performance, with NFC-Net typically surpassing CNN. Their accuracies indicate that they are competent models but may not possess all the specialized capabilities of the suggested model. ANN, DNN+HMM, and AlexNet models exhibit decent performance. Although they are somewhat effective at recognizing scripts, their lower accuracy results suggest limits, potentially in dealing with the complexities of Assamese script. When combined, LSTM and CNN show diverse performance but exhibit outstanding accuracy in certain queries such as Q35 and Q43. LSTM+CNN excels at specific script recognition tasks, perhaps due to LSTM's proficiency in processing sequential input and CNN's image analysis capabilities.

Table 8. Performance evaluation with different classifiers in terms of accuracy

Query Images	Proposed	NFC-Net	CNN	ANN	DNN+HMM	AlexNet	LSTM+CNN
Q1	0.99	0.93	0.91	0.88	0.87	0.84	0.8
Q5	0.97	0.88	0.86	0.83	0.78	0.79	0.75
Q10	0.98	0.9	0.88	0.85	0.87	0.82	0.84
Q13	0.96	0.87	0.85	0.82	0.81	0.79	0.75
Q15	0.94	0.89	0.87	0.84	0.83	0.81	0.78
Q20	0.92	0.85	0.83	0.8	0.79	0.77	0.73
Q23	0.95	0.9	0.87	0.84	0.83	0.81	0.78
Q25	0.93	0.83	0.81	0.78	0.77	0.75	0.71
Q30	0.96	0.93	0.91	0.88	0.87	0.85	0.81
Q33	0.94	0.88	0.86	0.82	0.81	0.79	0.75
Q35	0.98	0.9	0.89	0.84	0.83	0.95	0.94
Q40	0.95	0.87	0.9	0.8	0.79	0.9	0.88
Q43	0.92	0.91	0.85	0.85	0.86	0.91	0.91
Q45	0.96	0.88	0.88	0.83	0.83	0.88	0.87
Q50	1	0.95	0.94	0.92	0.96	0.93	0.98

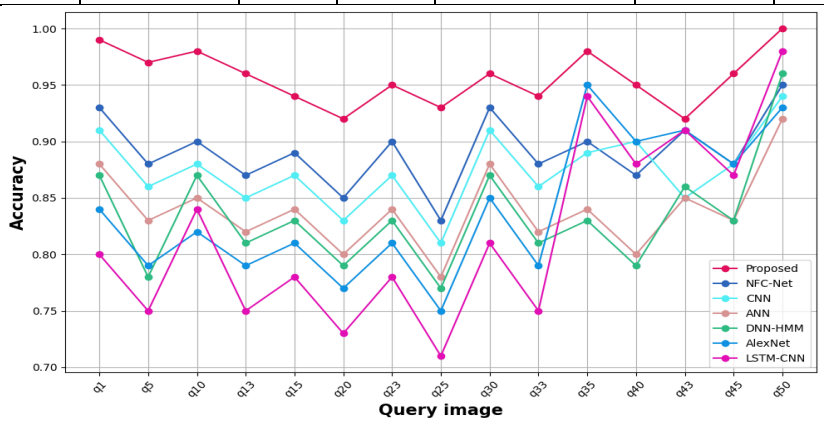


Figure 13. Analysis of Accuracy

Table 9 and Figure 14 display the precision analysis comparing the proposed method with existing methodologies. Precision is the ratio of relevant instances to the total number of retrieved instances. The Proposed Model is notable for its outstanding accuracy, demonstrating its effectiveness in precisely identifying important Assamese scripts with little false positives. NFC-Net and CNN models are powerful candidates, demonstrating their capacity to provide accurate results, although they might be further optimized to improve precision. The varying performance of LSTM+CNN in some queries underscores the significance of grasping the capabilities and constraints of each model, according to the script recognition task.

Table 9. Performance evaluation with different classifiers in terms of precision

Query Images	Proposed	NFC-Net	CNN	ANN	DNN+HMM	AlexNet	LSTM+CNN
Q1	0.95	0.9	0.89	0.88	0.87	0.84	0.8
Q5	0.9	0.86	0.84	0.86	0.83	0.79	0.75
Q10	0.92	0.84	0.88	0.83	0.8	0.82	0.84
Q13	0.94	0.8	0.82	0.85	0.74	0.79	0.75
Q15	0.91	0.85	0.85	0.82	0.78	0.81	0.78
Q20	0.93	0.8	0.73	0.84	0.87	0.77	0.73
Q23	0.9	0.86	0.78	0.8	0.81	0.81	0.7
Q25	0.92	0.78	0.8	0.84	0.83	0.75	0.78
Q30	0.94	0.8	0.76	0.78	0.79	0.85	0.71
Q33	0.9	0.86	0.8	0.88	0.83	0.79	0.81
Q35	0.92	0.82	0.88	0.82	0.77	0.84	0.75
Q40	0.96	0.86	0.82	0.86	0.87	0.8	0.78
Q43	0.98	0.9	0.86	0.9	0.81	0.9	0.82
Q45	0.92	0.93	0.9	0.95	0.88	0.86	0.76
Q50	1	0.95	0.94	0.97	0.95	0.96	0.94

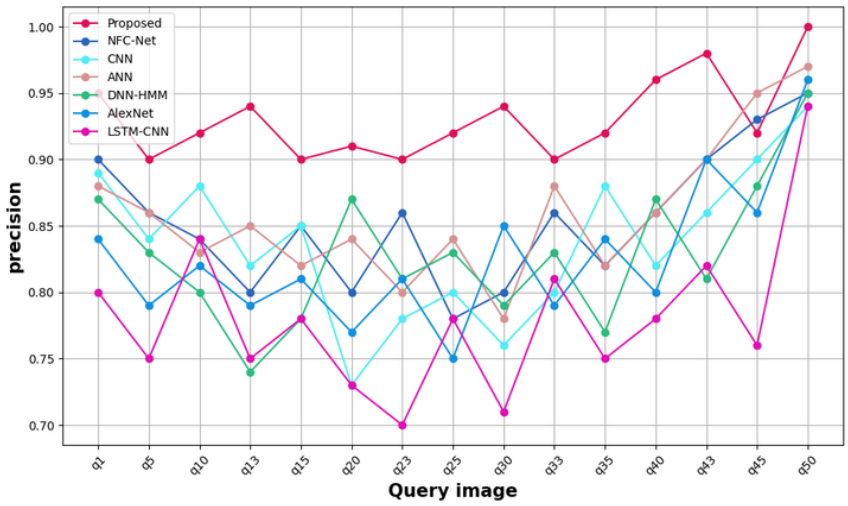


Figure 14. Analysis of precision

Table 10 and Figure 15 display the examination of recall comparing the suggested method with existing techniques. Recall, or sensitivity, quantifies the percentage of true positive cases properly recognized by the model. The proposed model has outstanding recall performance for all query photos, with scores between 0.92 and 1.00. The model's efficiency in obtaining relevant Assamese scripts indicates its suitability for applications where missing relevant scripts is unacceptable. The NFC+Net and CNN models exhibit good recall, even typically lower than the suggested model. They have shown the ability to recognize a large fraction of important scripts in their performance, although there are some cases where some relevant scripts are neglected. ANN,

DNN+HMM, and AlexNet exhibit decent recall performance. Their decreased recall relative to the top performers indicates that individuals may overlook more relevant scripts, which could be a limitation in applications requiring thorough retrieval. The LSTM+CNN model has diverse recall rates for different queries. Its performance indicates excellent effectiveness in specific cases but may not consistently retrieve all relevant scripts.

Table 10. Performance evaluation with different classifiers in terms of recall

Query Images	Proposed	NFC-Net	CNN	ANN	DNN+HMM	AlexNet	LSTM+CNN
Q1	0.99	0.94	0.91	0.88	0.87	0.84	0.8
Q5	0.97	0.89	0.86	0.83	0.78	0.79	0.75
Q10	0.98	0.92	0.88	0.85	0.87	0.82	0.84
Q13	0.96	0.89	0.85	0.82	0.81	0.79	0.75
Q15	0.95	0.92	0.87	0.84	0.83	0.81	0.78
Q20	0.94	0.87	0.83	0.8	0.79	0.77	0.73
Q23	0.95	0.93	0.87	0.84	0.83	0.81	0.78
Q25	0.93	0.86	0.81	0.78	0.77	0.75	0.71
Q30	0.96	0.9	0.91	0.88	0.87	0.85	0.81
Q33	0.94	0.91	0.86	0.82	0.81	0.79	0.75
Q35	0.96	0.93	0.89	0.84	0.83	0.83	0.8
Q40	0.92	0.9	0.9	0.8	0.79	0.8	0.78
Q43	0.96	0.95	0.85	0.85	0.86	0.82	0.83
Q45	0.98	0.92	0.88	0.83	0.83	0.86	0.9
Q50	1	0.98	0.94	0.92	0.96	0.9	0.92

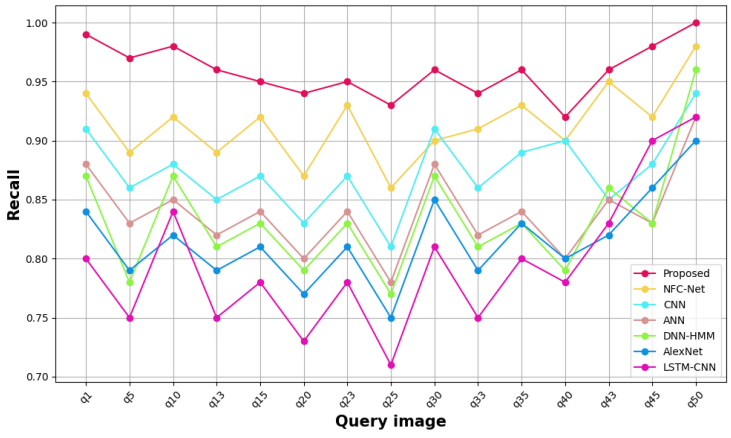


Figure 15. Analysis of Recall

Table 11 and Figure 16 display the study of F1 score comparing the suggested method with existing techniques. The F1 score is a statistical metric utilized to assess the precision of a binary classification system. The score is calculated by taking into account both the precision and recall of the test. The proposed model demonstrates outstanding F1 scores for all query photos, ranging from 0.93 to 1.00. The model has a strong convergence between precision and recall, indicating its high effectiveness in precisely and thoroughly recognizing Assamese script. NFC-Net

and CNN models exhibit respectable F1 results, even typically inferior to the suggested model. Their performance suggests a decent trade-off between precision and recall,however it may not outperform the proposed model in all aspects of script recognition. ANN, DNN+HMM, and AlexNet demonstrate reasonable F1 scores. Although effective to some degree, their lower F1 ratings compared to the best performers imply a potential trade-off between precision and recall, highlighting the need for enhancement. The combination of LSTM and CNN yields diverse F1 scores for distinct queries. Its result indicates the ability to strike a decent balance between precision and recall in specific contexts, albeit it may lack consistency across other script recognition tasks.

Table 11. Performance evaluation with different classifiers in terms of F1 score

Query Images	Proposed	NFC-Net	CNN	ANN	DNN+HMM	AlexNet	LSTM+CNN
Q1	0.97	0.9	0.8	0.76	0.81	0.78	0.76
Q5	0.96	0.85	0.83	0.79	0.86	0.8	0.79
Q10	0.98	0.86	0.85	0.77	0.89	0.81	0.78
Q13	0.93	0.87	0.89	0.79	0.82	0.8	0.79
Q15	0.96	0.88	0.82	0.8	0.82	0.84	0.8
Q20	0.95	0.85	0.88	0.76	0.85	0.86	0.79
Q23	0.98	0.89	0.9	0.79	0.89	0.85	0.84
Q25	0.99	0.9	0.93	0.8	0.87	0.88	0.82
Q30	0.95	0.93	0.91	0.83	0.9	0.8	0.89
Q33	0.94	0.9	0.87	0.8	0.93	0.85	0.86
Q35	0.96	0.88	0.9	0.82	0.91	0.9	0.9
Q40	0.98	0.85	0.92	0.91	0.9	0.93	0.93
Q43	0.99	0.89	0.96	0.89	0.93	0.9	0.91
Q45	0.99	0.9	0.95	0.9	0.95	0.94	0.94
Q50	1	0.92	0.96	0.93	0.96	0.95	0.97

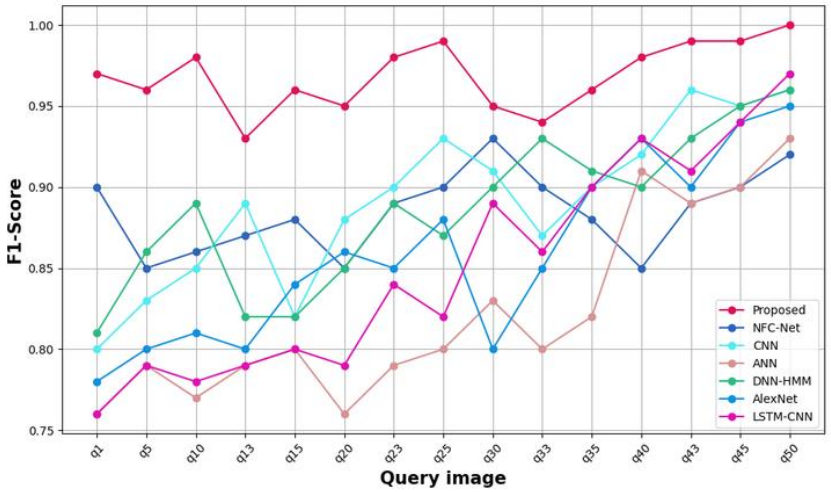


Figure 16. Analysis of F1 score

Table 12 and Figure 17 display the kappa analysis results comparing the suggested method with existing techniques. The Kappa statistic, often known as Cohen's Kappa, quantifies the level of

agreement amongst raters when assessing categorical items. This is a method used to measure the degree of agreement between two raters (or in machine learning, between an algorithm's predictions and the actual values) while accounting for random agreement. The proposed model demonstrates outstanding Kappa scores for all query photos, ranging from 0.90 to 1.00. This indicates that the model's predictions closely match the actual classifications of the Assamese scripts, beyond what would be anticipated by random chance. This suggests a very dependable strategy for recognizing Assamese writing. NFC-Net and CNN models have respectable Kappa scores, although typically inferior to the suggested model. Their performance shows a significant level of agreement with the actual script classifications, indicating that they are competent models but may not be as accurate or consistent as the suggested model. ANN, DNN+HMM, and AlexNet exhibit moderate Kappa scores. Although showing a moderate level of agreement above random chance, their lower scores in comparison to the top performers imply greater unpredictability in their predictions or less accuracy in aligning with the genuine classifications. The combination of LSTM and CNN yields diverse Kappa values for distinct queries. Its results indicates that it can obtain a high level of compatibility with actual script classifications in specific situations, but it may not be consistently dependable for other types of script recognition work.

Table 12. Performance evaluation with different classifiers in terms of Kappa

Query Images	Proposed	NFC-Net	CNN	ANN	DNN+HMM	AlexNet	LSTM+CNN
Q1	0.99	0.93	0.81	0.88	0.87	0.84	0.8
Q5	0.95	0.83	0.76	0.86	0.73	0.79	0.75
Q10	0.96	0.85	0.87	0.83	0.82	0.82	0.84
Q13	0.93	0.82	0.8	0.85	0.75	0.79	0.75
Q15	0.91	0.84	0.84	0.82	0.8	0.81	0.78
Q20	0.9	0.8	0.7	0.84	0.75	0.77	0.73
Q23	0.95	0.85	0.77	0.8	0.79	0.81	0.7
Q25	0.93	0.83	0.75	0.84	0.74	0.75	0.78
Q30	0.95	0.93	0.8	0.78	0.84	0.85	0.71
Q33	0.94	0.83	0.72	0.88	0.77	0.79	0.81
Q35	0.96	0.88	0.85	0.82	0.79	0.82	0.75
Q40	0.95	0.87	0.82	0.86	0.79	0.76	0.78
Q43	0.93	0.91	0.86	0.9	0.86	0.8	0.82
Q45	0.96	0.88	0.9	0.95	0.83	0.88	0.76
Q50	1	0.95	0.94	0.97	0.96	0.95	0.94

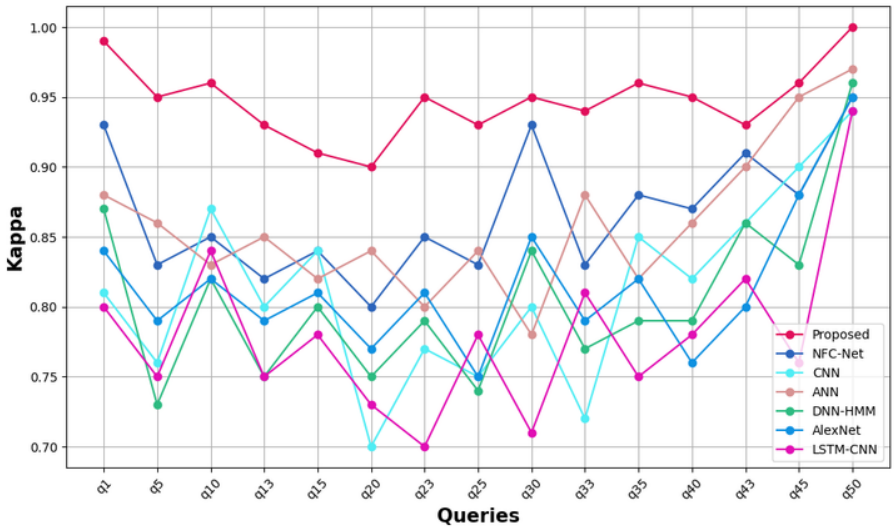


Figure 17. Analysis of Kappa

Figure 18 presents a precision-recall graph that evaluates the performance of various machine learning models in the task of Assamese script recognition within PDF documents. The proposed model, demonstrates superior performance across the range of recall values when compared to the other models, including NFC-Net, a Convolutional Neural Network (CNN), AlexNet, and an LSTM-CNN hybrid. Notably, the proposed model maintains a higher level of precision at lower recall levels, indicating a robust ability to retrieve relevant instances of Assamese script without incurring a significant number of false positives. The consistency of the proposed model's performance, with minimal drops in precision as recall increases, suggests an efficient balance between retrieving as many relevant instances as possible (recall) and ensuring that the retrieved instances are indeed relevant (precision).

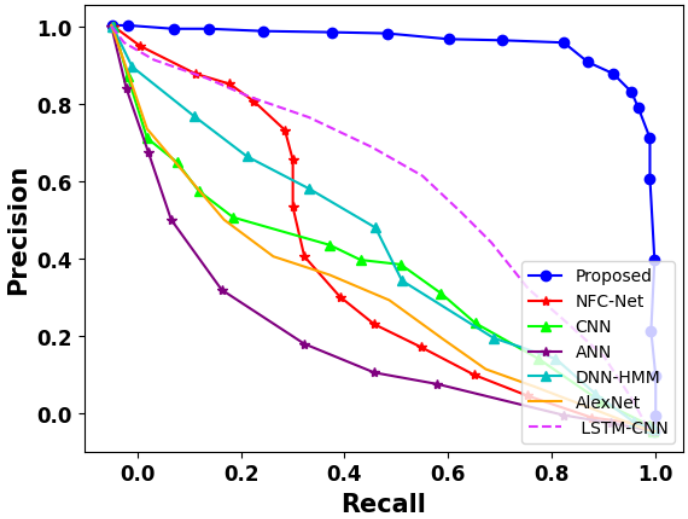


Figure 18. Precision-Recall Performance Comparison

4.7 Analysis of Retrieval performance

The experiment yields images with 20650 words. In 110 printed pages, a total of 270 instances of more than 70 terms are utilized as query words during the retrieval session. The outcome of the suggested strategy work on the retrieval session is displayed in Table 13.

Table 13. Retrieval results of the proposed approach

Query Image	এজনী	উভতি	জানিবা
Retrieved Page	ক'লত বুটীয়ে মাত লগালে, "বোপা ঐ-টেঙাটোৰ ভিতৰতে তোমাৰ কুঁৱৰী আছে, গোৱালি-ঘৰৰ ভিতৰত একুৰা টুৰুজুই ধৰি থবা আৰু দৈ গাখীৰ আঠিয়াকল হেনেকি দিয়লগই এগৰাকে এটা বাটিত দৈ দিবা। তুমি তোমাৰ গোৱালগাটিত টোপনি অহাৰ ভাও জুৰি নাক ঘোৰোঁৱাই থাকিবা। তোমাৰ টোপনি আহিছে ভাবি যেতিয়া সেই ঐ-টেঙাটোৰ পৰা ছোৱালীজনী ওলাই ভাত খাবলৈ বহিব যেতিয়া তুমি তত্তালিকে উঠি সেই ঐ-টেঙাটো টুৰুজুইত দি পুৰি পেলাবা। কিন্তু যেতিয়া দেখিবা যে ছোৱালীজনী চীচ কৰে মাটিত পৰি গৈছে। যেতিয়াই লৰালৰিকৈ সেই বাটিখনৰ পৰা গাখীৰ আৰু আঠিয়াকল আনি তাই সুধত ডৰুৱা দিবা, দিলেই তাই জান পাই উঠি বহিব। বুটীৰ বুধি শুনি কোঁৱৰৰ বৰ পাই চাইক বাঁচা দি দিয়াম দিলে; আৰু কাতি সেই বুটীৰ কথামতেই কৰি ঐ-কুঁৱৰীক লাভ কৰিলে। কোঁৱৰৰ বাপেক-মাকে পিছদিনা পুৱা এই ঘটনা শুনি আৰু বোৱাৰীয়েকক দেখি তেওঁলোকৰ মনত আনন্দই নধৰা হ'ল। বিয়ে-বিয়েনী ৰঙা-কাপৌলে এই বাতৰি তেওঁলোকে কৈ পৃথিবীলত তেওঁলোকেৰা মনত মহা ৰং মিলিল আৰু আকৌ তেওঁলোকে মহা পয়োভৰেৰে জীয়েক ঐ-কুঁৱৰীক কোঁৱৰলৈ বিয়া দিলে। ****** সমাপ্ত ****** টোঁৰাকাউৰী আৰু টিপচীচৰাই এক বোলে টোঁৰাকাউৰী আৰু এটা টিপচীচৰাই আছিল। সিহঁত দুয়ো সখি বস্তালে। এদিন দুয়ো একেলগে এজোপা গছৰ তলত বহি আছিল। এনেতে দেখিলে যে এজনী বুটীয়ে ধান সিজাই চোতালত মেলি দিছে। টোঁৰাকাউৰীয়ে টিপচীক ক'লে, 'সখি, সখি চোৱাচোন সেই বুটীজনীয়ে সিজোৱা ধান ৰ'দত মেলি দিছে, এথান মাৰি আনি আলেচিক আহোঁ। টিপচীয়েও 'এৰা আলেচিক আহোঁ' বুলি মাত দিলত টোঁৰাকাউৰীয়ে ক'লে, "কিন্তু সখি এটা কথা, মই ঠোঁতত যিমান ধান আনিম তুমিও যদি সিমান আনিব লোৱাৰা তেনেহ'লে মই তোমাক খাম। আৰু তুমি যিমান আনিবা ময়ো যদি সিমান আনিব নোৱাৰোঁ তেনে তুমি মোক খাবা।" টোঁৰাকাউৰীৰ কথা শুনি টিপচীয়ে ভাবিলে, "মইতো ভাতকৈ বেছি ধান একে থাপতে আনিব নোৱাৰিমেই, সেইদেখি বুজিছোঁ সি মোক খাবলৈ বুধি পাতিছে। ৰাক যেখাই থাওকচোন সি মোক কেনেকি খাম।" ইয়াকে ভাবি টিপচীয়ে উত্তৰ দিলে, "ৰাক সখি সেয়ে হওক।" টোঁৰাকাউৰীয়ে ক'লে, "তেন্তে তুমি আগেয়ে থাপ মাৰি আনা।" টিপচীয়ে উত্তৰ দিলে, "নহয় সখি তুমি আগেয়ে আনা।" কাউৰীয়ে ক'লে, "তুমিয়ে আগ হ'ব লাগে মই তোমাৰ পিছতহে।" এই কথা শুনি টিপচীয়ে তাৰ এথুমহান ঠোঁটেৰে থাপামাৰি এঠোঁত ধান আনিলে; কিন্তু সি নিচেই তাকৰা। তাৰ পিছত টোঁৰাকাউৰীয়ে থাপ মাৰি ইমান ধান আনিলে যে	অলপ পা-বেয়া হোৱা বাবে মাকে মঙ্গলডী বুটী এজনী মাতি আনি মঙ্গল চোৱালে। বুটীয়ে মঙ্গল চাই ক'লে যে কিন্তুমান জাতি ভকতক পন্থৰে ভোজ এটা যুৱালে কাঁৰী-কোঁৱৰক পা তাল হ'ব। এই কথা শুনি কাঁৰীয়ে পুতেকক ক'লত পুতেকে বাঘক ভোজনৈ দুটা পহু আনি দিবলৈ ক'লে। বাঘে ভোজৰ দিনা দুটা পহু মাৰি আনি কাঁৰী-কোঁৱৰক দি গ'ল। ভোজৰ দিনা সকলো জাতি কুটুমে কাঁৰী-কোঁৱৰৰ ঘৰত খাবলৈ বহিছে, এনেতে এখন পাত কম পৰাত এজনে কলপাত এখন কাটি আনিবৰ মনেৰে বাৰীৰ পাছফালেৰ দৈ দেখিলে কিহবাৰ হাড়ে মাস লগোৱাকাম দৈ আছে। মানুহটোৱে হাড়বোৰ দেখি বিচুৰ্তি খাই পাত নাকাটি উভতি গুচি আহি সকলোকে ক'লে, "ৰাইজকলৰ, আমি ইয়াৰ ঘৰত ভোজ খাব নোৱাৰোঁ, ই কিবা খাদ্য। আপোনাসকলে আহি চাওকচোন। ই মানুহবোৰ হাড় ন'মাই ৰ'বলৈ ক'ত পালে?" এইবুলি ক'লত, সকলো ভোজৰ ভাতৰ পৰা হাত দাঙি উঠি আহি দেখিলে কথা সঁচা। তেতিয়া সকলোৱে কাঁৰী-কোঁৱৰক সুধিলে, "এইবোৰ কিহৰ হাড়-মুৰ ক? নহ'লে আমি তোৰ ঘৰত ভোজ খাব নোৱাৰোঁ।" কাঁৰী-কোঁৱৰে ক'লে, "খাব পোৱা জন্তুৰ বাহিৰে আন একোৰে হাড়-মুৰ নহয়। আপোনাসকলৰ আগত ক'ব নোৱাৰো; কাৰণ ক'লেই মোক বাঘে দৈ গুচি যাবহি। সেইবাবে আপোনাসকলে খুকুৰি নাবাধি মোক ক্ষমা কৰি ভোজ-ভাত খাওকা।" বাইজে তাৰ কথা শুনি নপতিয়াই তাক ক'লে, "আমি সকলোৱে যাচি-জোৰ দৈ তোক বেচি থাকোঁ, ভই ক; নহ'লে আমি নাখাওঁ।" বাইজৰ কথা শুনি কাঁৰী-কোঁৱৰে নিকপায় হৈ সকলোৰে মাজত আঁঠু দৈ কথাটো ক'বলৈ ধৰিলে। সকলোৱে যাচি জোৰ দৈ তাক বেচি ৰ'ল। যেই কথাটো শেষ হ'ল বিধাপৰলে ক'ৰবাৰ পৰা বাঘটো আহি মানুহৰ মাজৰ পৰা তাক থাপ মাৰি ধৰি দৈ গুচি গ'ল। ভোজ খাবলৈ অহা ৰাইজে অলপমান বেলি ভেবালাগি তাৰ পিছত ইফালে-সিফালে তাক নাপাই ঘৰঘাবৰি গুচি গ'ল। ইপিনে বাঘে কাঁৰী-কোঁৱৰক নি এথন লেজেপতি হাবিত সোমালেগৈ। কোঁৱৰৰ কিবা ভাগ্য ভাল আছিল, এনে সময়তে ভোকত তাৰ পেটে কোকোঁৱাবলৈ ধৰিলে। বাঘে কোকোঁৱৰ শুনি কোঁৱাক সুধিলে, 'তোৰ পেতত কিয়ে কোকোঁৱাছে অ'?" কথাষাৰ শুনি কাঁৰী-কোঁৱৰৰ খপৰবে বুধি ওলাল। সি উত্তৰ দিলে, "আয়ে সৰুতে মোক ঘুকুৰি কৈকোৱা খুৱাইছিল, তাৰে তিনিকুৰি আগেয়ে ওলাল, তিনিকুৰি বাকী আছিল। এতিয়া সেই তিনি-কুৰিয়েই ওলাব খুজিছে।" বাঘে কৈকোৱাৰ নাম শুনিয়েই ভাবিলে-"সৰ্বনাশ। এটা কৈকোৱাই মোক চৈধ্য ভুৰক দেখুৱাই থৈছে। এতিয়া যদি তিনিকুৰি কৈকোৱা ওলাব লাগিল, তেনেহ'লে আমাৰ ফালে ছৈছে আৰু!" বাঘে ইয়াকে ভাবে কাঁৰী-কোঁৱৰক তাতে এৰি থৈ লৰি পলাই গুচি গ'ল। কাঁৰী-কোঁৱৰে বাঘৰ হাতৰ পৰা এৰাই লাহে লাহে দৈ এখন বৈৰ পাৰত এজোপা গছৰ তল পালেগৈ। সি দেখিলে, গছজোপাৰ তলতে বনৰীয়া ম'ত বুটী। ম'হবোৰ তেতিয়া তাত নাহিল দেখি কোঁৱৰে লৰালৰিকৈ তাত পৰি থকা গাখীৰৰ ফেনবোৰ খাই বুটিখন ডালকৈ সাৰি-মচি চাফ-চিকুণ কৰি থৈ ওচৰৰ	পুত্ৰ কন্যা পৰিবাৰ কোন তযু তুমি কাৰ বুলি হেৰা অবাবত নকৰী বেজাৰ; কলা-ঘুমটিয়া হই, আতমা যে আছে বই, জানিবা মৰণ নাই এই আতমাৰ। ৩ অস্থায়ী মানৰ দেহা, কৰিম দুদিন বেহা, ফুৰিম দুদিন মাথোঁ সংসাৰ-ঘাটত; দুদিনৰ অন্ত হাল, আয়ুষ চুকাই গাল, সুন্দৰ দেহাৰ ঠাই শেষ শ্মশানত। ৪ চকু কাণ নাক যাব, হাড় ছাল নাশ পাব, মাটিৰ মানুহ তুমি মাটিত মিলিবা; অবিনাশী নিত্য ধন, অমৰণ অভগন, অনন্ত উন্নতিশীল আতমা জানিবা! ৫

From table 14, the query instance under consideration yielded a total of 375 potential script matches. Out of these, the system correctly identified 360 instances as true positives, indicating that these matches were relevant and accurately recognized as instances of Assamese script. The system incurred 6 false positives, which are instances that were incorrectly identified as relevant. Additionally, there were 9 false negatives, which represent relevant instances that the system failed to identify.

Table 14. Retrieval results of proposed approach

Query instance	True positive	False positive	False negatives	Recall	Precision
375	360	6	9	98.20%	98.12%

Some of the retrieval words are incorrect. After examining the wrongly recovered words, it was discovered that the improper retrieval result was caused by some words that were only varied by one character, and the shape of the character was very similar, as well as the extract feature points of the query word and test word imagine being the same.

The suggested method's performance is evaluated in terms of recognition rate, retrieval time, and error parameters. In addition, the suggested technique is compared to several classifiers such as NFC-Net, CNN, ANN, DNN+HMM, AlexNet, and LSTM+CNN (Table 15). The suggested classifier achieves a high recognition rate of 98.7% for Assamese text identification. In CBIR, the time required to retrieve similar images is an important element to consider. Extracting the most characteristics may result in additional computation and retrieval time because the amount of features and retrieval time are inversely proportional. The proposed technique strikes a satisfactory compromise between retrieval time and retrieval accuracy. Furthermore, the suggested approach has error rates in the region of 1.3% for Assamese script recognition. The proposed strategy achieves high accuracy, precision, recall, F-measure, Kappa,

and additionally the mistake rates are lower when compared to existing techniques for successfully recognizing the words.

Table 15. Performance evaluation with different classifiers

Techniques	Recognition Rate (%)	Retrieval time (sec)	Error (%)
Proposed	98.7	35.12	1.3
NFC-Net	94.5	75.87	5.5
CNN	96.2	68.14	3.8
ANN	94.12	82.65	5.88
DNN+HMM	87.6	99.47	12.4
AlexNet	92.54	114.57	7.46
LSTM+CNN	96.8	107.54	3.2

5. Conclusion and future works

The authors have proposed a character recognition system for Assamese script papers that extracts text from an image and then searches for the whole document containing the content. The linguistic content of the image was used as the foundation. The current technology can be used to store and retrieve picture documents. Essentially by giving a query image, the User will be directed to the appropriate document in the database. This technique can be used in instances in which documents need to be scanned into images for later retrieval based on a query image. The system effectively prepares PDF document images for subsequent processing by incorporating a series of methodical preprocessing stages, such as grayscale conversion and adaptive thresholding. These stages provide binarized images that clearly distinguish the script from the backdrop, which is required for accurate character recognition. The system's process, as shown in the flowchart, offers a robust strategy that includes the training and query stages, with neural networks such as the ASRNN and the Inception v3 model used for word segmentation and mapping, respectively. The ISNN is crucial in extracting shape-based features, which are required for the similarity matching process. The results are equivalent to previous methods. Performances are measured using a variety of criteria, including average accuracy rate, kappa statistics, error rate, TP rate, FP rate, precision, recall, and F-measure. We reported a recognition accuracy of 98.7%. Future study could focus on enhancing CNN architecture by experimenting with various layers, filters, and activation functions to improve feature extraction and recognition accuracy. Furthermore, investigating transfer learning by employing pre-trained models on similar scripts and fine-tuning them for the Assamese script could be advantageous.

References

[1] Garg, M., & Dhiman, G. (2021). A novel content-based image retrieval approach for classification using GLCM features and texture fused LBP variants. *Neural Computing and Applications*, *33*(4), 1311-1328.

[2] Ahmad, F. (2022). Deep image retrieval using artificial neural network interpolation and indexing based on similarity measurement. *CAAI Transactions on Intelligence Technology*, *7*(2), 200-218.

[3] Alshehri, M. (2020). A content-based image retrieval method using neural network-based prediction technique. *Arabian journal for science and engineering*, *45*(4), 2957-2973.

[4] Fachruddin, F., Saparudin, S., Rasywir, E., &Pratama, Y. (2022). Network and layer experiment using convolutional neural network for content based image retrieval work. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, *20*(1), 118-128.

[5] Kundu, S., Paul, S., Singh, P. K., Sarkar, R., &Nasipuri, M. (2020). Understanding NFC-Net: a deep learning approach to word-level handwritten Indic script recognition. *Neural Computing and Applications*, *32*, 7879-7895.

[6] Ghosh, S., Chatterjee, A., Sen, S., Kumar, N., & Sarkar, R. (2021). CTRL–CapTuRedLight: a novel feature descriptor for online Assamese numeral recognition. *Multimedia Tools and Applications*, *80*, 30033-30056.

[7] Medhi, K., & Kalita, S. K. (2020). Assamese Handwritten Character Recognition using Supervised Fuzzy Logic. *Int. J. Recent Technol. Eng*, *8*(5), 3750-3758.

[8] Choudhury, A., & Sarma, K. K. (2021). A CNN-LSTM based ensemble framework for in-air handwritten Assamese character recognition. *Multimedia Tools and Applications*, *80*(28), 35649-35684.

[9] Zhao, L., & Tang, J. (2010, October). Content-based image retrieval using optimal feature combination and relevance feedback. In *2010 International Conference on Computer Application and system modeling (iccasm 2010)* (Vol. 4, pp. V4-436). IEEE.

[10] Ghahremani, M., Ghadiri, H., &Hamghalam, M. (2021). Local features integration for content-based image retrieval based on color, texture, and shape. Multimedia Tools and Applications, 80(18), 28245-28263.

[11] Li, X., Yang, J., & Ma, J. (2021). Recent developments of content-based image retrieval (CBIR). *Neurocomputing*, *452*, 675-689.

[12] Hafemann, L. G., Oliveira, L. S., & Sabourin, R. (2018). Fixed-sized representation learning from offline handwritten signatures of different sizes. *International Journal on Document Analysis and Recognition (IJDAR)*, *21*, 219-232.

[13] Theetchenya, S., Ramasubbareddy, S., Sankar, S., & Basha, S. M. (2021). Hybrid approach for content-based image retrieval. *International Journal of Data Science*, *6*(1), 45-56.

[14] Maiwald, F., Lehmann, C., &Lazariv, T. (2021). Fully automated pose estimation of historical images in the context of 4D geographic information systems utilizing machine learning methods. *ISPRS International Journal of Geo-Information*, *10*(11), 748.

[15] Byju, A. P., Demir, B., & Bruzzone, L. (2020). A progressive content-based image retrieval in JPEG 2000 compressed remote sensing archives. *IEEE Transactions on Geoscience and Remote Sensing*, *58*(8), 5739-5751.

[16] Zhao, S., Wang, L., Qian, X., & Chen, J. (2022). Enhancing performance-based generative architectural design with sketch-based image retrieval: a pilot study on designing building facade fenestrations. *The Visual Computer*, *38*(8), 2981-2997.

[17] Mohite, N. B., & Gonde, A. B. (2022). Deep features based medical image retrieval. *Multimedia Tools and Applications*, *81*(8), 11379-11392.

[18] Hamreras, S., Boucheham, B., Molina-Cabello, M. A., Benitez-Rochel, R., & Lopez-Rubio, E. (2020). Content based image retrieval by ensembles of deep learning object classifiers. *Integrated computer-aided engineering*, *27*(3), 317-331.

[19] Ghahremani, M., Ghadiri, H., &Hamghalam, M. (2021). Local features integration for content-based image retrieval based on color, texture, and shape. Multimedia Tools and Applications, 80(18), 28245-28263.

[20] Alsmadi, M. K. (2020). Content-based image retrieval using color, shape and texture descriptors and features. *Arabian Journal for Science and Engineering*, *45*(4), 3317-3330.

[21] Varish, N. (2022). A modified similarity measurement for image retrieval scheme using fusion of color, texture and shape moments. *Multimedia Tools and Applications*, *81*(15), 20373-20405.

[22] Dubey, S. R. (2021). A decade survey of content based image retrieval using deep learning. *IEEE Transactions on Circuits and Systems for Video Technology*, *32*(5), 2687-2704.

[23] Du, W. S. (2021). Subtraction and division operations on intuitionistic fuzzy sets derived from the Hamming distance. *Information Sciences*, *571*, 206-224.

[24] Müller, H., Michoux, N., Bandon, D., & Geissbuhler, A. (2004). A review of content-based image retrieval systems in medical applications—clinical benefits and future directions. *International journal of medical informatics*, *73*(1), 1-23.

[25] Ballerini, L., Li, X., Fisher, R. B., & Rees, J. (2010). A query-by-example content-based image retrieval system of non-melanoma skin lesions. In *Medical Content-Based Retrieval for Clinical Decision Support: First MICCAI International Workshop, MCBR-CDS 2009, London, UK, September 20, 2009, Revised Selected Papers 1* (pp. 31-38). Springer Berlin Heidelberg.

[26] Rasheed, A., Ali, N., Zafar, B., Shabbir, A., Sajid, M., & Mahmood, M. T. (2022). Handwritten Urdu characters and digits recognition using transfer learning and augmentation with AlexNet. IEEE Access, 10, 102629-102645.

[27] Banerjee, D., Bhowal, P., Malakar, S., Cuevas, E., Pérez-Cisneros, M., & Sarkar, R. (2022). Z-transform-based profile matching to develop a learning-free keyword spotting method for handwritten document images. International Journal of Computational Intelligence Systems, 15(1), 93.

[28] Ghazal, T. M. (2022). Convolutional neural network based intelligent handwritten document recognition. Computers, Materials & Continua, 70(3), 4563-4581.

[29] Lakshmi, K. M. (2021). An efficient telugu word image retrieval system using deep cluster. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12(11), 3247-3255.

[30] Zagoris, K., Amanatiadis, A., &Pratikakis, I. (2021). Word spotting as a service: an unsupervised and segmentation-free framework for handwritten documents. Journal of Imaging, 7(12), 278.

[31] Wunnava, A., Naik, M. K., Panda, R., Jena, B., & Abraham, A. (2020). A novel interdependence based multilevel thresholding technique using adaptive equilibrium optimizer. *Engineering Applications of Artificial Intelligence*, *94*, 103836.

[32] Yang, P., Song, W., Zhao, X., Zheng, R., & Qingge, L. (2020). An improved Otsu threshold segmentation algorithm. *International Journal of Computational Science and Engineering*, *22*(1), 146-153.

[33] Peng, Z., Li, X., Zhu, Z., Unoki, M., Dang, J., & Akagi, M. (2020). Speech emotion recognition using 3d convolutions and

attention-based sliding recurrent networks with auditory front-ends. *IEEE Access*, *8*, 16560-16572.

[34] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-9).

[35] Wang, C., Chen, D., Hao, L., Liu, X., Zeng, Y., Chen, J., & Zhang, G. (2019). Pulmonary image classification based on inception-v3 transfer learning model. *IEEE Access*, *7*, 146533-146541.

[36] Subrahmanyeswara Rao, B. (2020). Accurate leukocoria predictor based on deep VGG-net CNN technique. *IET Image Processing*, *14*(10), 2241-2248.

[37] Lu, S., Lu, Z., & Zhang, Y. D. (2019). Pathological brain detection based on AlexNet and transfer learning. *Journal of computational science*, *30*, 41-47.

[38] Madduri, A. (2021). Content based Image Retrieval System using Local Feature Extraction Techniques. *International Journal of Computer Applications*, *183*(20), 16-20.

[39] Chicco, D. (2021). Siamese neural networks: An overview. *Artificial neural networks*, 73-94.

[40] Varish, N., Kumar, S., & Pal, A. K. (2017). A novel similarity measure for content based image retrieval in discrete cosine transform domain. *FundamentaInformaticae*, *156*(2), 209-235.