Devanagari Handwritten Character Recognition Using Neural Network

Anupama Thakur, Amrit Kaur

Abstract— Optical Character Recognition is a framework which can translate the images from manually handwritten or printed structure to machine-editable structure. Devanagari script is utilized in numerous Indian dialects like Hindi, Nepali, Marathi, Sindhi and so on. This script structures the establishment of the language like Hindi which is the national and most generally communicated language in India. In current scenario, there is a tremendous interest of accumulating the data in advanced configuration accessible in paper archives and after that later reusing this data by a search procedure. In this paper, we propose a new strategy for recognition of printed Hindi characters in Devanagari script. In this research, the main focus is given towards the recognition of the individual consonant and vowel which can be later reached out to perceive complex inferred words. In this undertaking the fundamental accentuation is given towards the recognition of the individual consonant and vowel. In this project, different pre-processing operations like features extraction, segmentations and classification methods have been studied and implemented to design a sophisticated OCR system for Hindi. In previous research, the classification K-NN technique have been implemented, but in proposed work, we have used hybrid technique which contains k-NN along with neural networks. Proposed approach provides 97.4% recognition rate as compared to 94.5% for existing techniques, which indicates that the proposed approach is better as compared to the techniques used in existing method.

Index Terms— Devanagari, OCR, *k*-NN, Neural Network, BPNN, SVM.

1 INTRODUCTION

TOptical character recognition or optical character per user, frequently curtailed as OCR, is the mechanical or electronic conversion of pictures of composed, written by hand or printed content into machine-encoded content, regardless of whether from a filtered archive, a photograph of a report, a scene-photograph (for instance the content on signs and boards in a scene photograph) or from caption content superimposed on a picture. Generally utilized as a type of data section from printed paper information records – regardless of whether international ID archives, solicitations, bank proclamations, modernized receipts, business cards, mail, printouts of static-information, or any reasonable documentation – it is a typical strategy for digitizing printed messages with the goal that they can be electronically altered, looked, stored more minimally, showed -line, and utilized in machine procedures, for example, psychological figuring, machine interpretation, (extracted) content-to-speech, key information and content mining. OCR is an area of research in artificial intelligence, computer vision and pattern recognition. Handwriting recognition has innumerable applications in the field of image processing and pattern recognition like authentication of signatures in banks, processing of archived institutional records, recognizing ZIP code addresses on letters, forensic evidence, etc. It has the potential to advance automation process and can reduce human intervention in numerous applications. Several techniques and methods are being developed to achieve higher accuracy with reduced processing time. Handwritten recognition is of offline and online type. In the on-line recognition system, the two-dimensional co-ordinates of successive points of the writing are stored sequentially as a function of time. In the off-line recognition, only the completed writing captured by an optical scanner is available as an image. Hence, the online method shows higher rate of recognition than the off-line method due to the availability of temporal information [1]. However, several applications require off-line handwriting recognition systems for complete automation and as a result, it continues to be an active area of research.

2 DEVANAGARI SCRIPT

Devanagari script is the establishment of numerous Indian dialects like Hindi, Nepali, Marathi, Sindhi and so forth and utilized by in excess of 300 million individuals around the globe. So Devanagari script assumes an exceptionally real job in the improvement of writing and compositions. There is such an extensive amount writing from the maturity compositions, Vedas and sacred texts and since these are so old so these are not effectively open to everybody. The need and desire to peruse these seniority sacred texts prompted the advanced change of these by examining the books. For checking and changing over the archives into editable structure OCR framework for Devanagari content was presented. This editable structure out of yield content can be input to different frameworks like it very well may be orchestrated with the voice to hear the charm of sacred texts and so on. Devanagari content is written in left to right arrangement [2]. It comprises of 11 vowels and 33 essential consonants. Every vowel except the initial one have comparing modifier utilizing which we can adjust a consonant. This line which is accessible in the upper side of a character is classified "Shirorekha". In view of this shirorekha each character is separated into three particular parts. The segment in the upper side of shirorekha is called upper modifiers, in the center segment the character is accessible and in the last part lower modifiers are accessible. Additionally, a few characters consolidate to frame another character set called joint characters. Example is shown in figure 1.

Fig1: Three strips of a word [3]

3 Challenges of Character Recognition

Machine-printed text includes the materials such as books, newspapers, magazines, documents and various writing units in the video or still images. The problem of recognition of fixed-font, multi-font and omni-font character is relatively well understood and solved with some constraints. Documents generated on a high quality paper with modern printing technologies allow the systems to exceed 99% recognition accuracy. However, the recognition rate of the commercially available products depends on the age of the documents, quality of the paper and ink, which may result in significant data acquisitions noise. Documents with coloured or patterned backgrounds, marked with pens, crooked when scanned, can yield poor OCR results [4]. Some improvements can be done by either adjusting the scanner settings and rescanning the document or manually correcting the electronic data.

4 Traditional Techniques

4.1 Problem domain reduction techniques

Narrowing the problem domain often helps increase the accuracy of handwriting recognition systems. A form field for a U.S. ZIP code, for example, would contain only the characters 0-9. This fact would reduce the number of possible identifications.

Primary techniques:

- Specifying specific character ranges
- Utilization of specialized forms

4.2 Character extraction

Off-line character recognition often involves scanning a form or document written sometime in the past. This means the individual characters contained in the scanned image will need to be extracted. Tools exist that are capable of performing this step [5].

4.3 Character recognition

After the extraction of individual characters occurs, a recognition engine is used to identify the corresponding computer character. Several different recognition techniques are currently available.

5 Limitations of Character Recognition

OCR for Indian Languages has never achieved a read rate that is 100%. Because of this, a system which permits timely and accurate, correction of rejects is a major requirement. Of even greater concern is the problem of misreading a character (substitutions). The success of any character recognition system is to read accurately the characters without substitutions. Visual images are also subject to noise and therefore, there are issues particularly over edges [6]. This leads to loss of accuracy and the system doing recognizing predicts characters with low accuracy adding further problems in the subsequent steps. The limitations of this work are as follows:

1. Specially designed fonts (characters) are not considered.
2. Preformatted paper is used to collect the data samples.
3. Mathematical model alone being considered for recognition.

6 Limitations Of Traditional Techniques

Existing Scheme utilized two new calculations are proposed for character division and recognition for multilingual Indic archives comprising of printed and written by hand messages. Character division is one of the significant strides before OCR. Thus, heuristic based calculation incorporated with SVM is proposed for fragmenting characters. Covered and over fragmented characters are likewise isolated precisely during post-preparing and post-check stages, separately.

Most noteworthy SR of 98.86% is gotten on exclusive database of Latin content. For character recognition, three new auxiliary geometry based highlights are proposed. FCDF and FCCF are determined regarding focus pixel of diminished character picture, while NCF is determined utilizing neighborhood data of content pixels. Proposed recognition calculation indicates most noteworthy exactness of 99.84% on Chars74k numerals database.

Low Accuracy in Devanagari Script.

Higher Complexity.

Low accuracy in alphabets rather than numerals.

7 Block Diagram

Any character recognition system goes under following steps, i.e. Image acquisition, Preprocessing, Segmentation, Feature extraction, classification and post processing. Block diagram of general character recognition system is shown in Figure 2.

Scanning: First characters composed on the printed copy record is get examined by scanner and after that picture is changed over into jpg format.

Pre-processing: Pre-preparing includes following advances. In the proposed OCR framework, content is digitized with the assistance of scanner having goals somewhere in the range of 100 and 600 dpi. The digitized pictures are as a rule in dim tone [1].

Binarization: This stage comprise of the way toward changing over a dim scale picture into twofold picture by thresholding. Two power esteems are acquired as Black and White.

Size normalization: As transcribed characters are not uniform in size. Along these lines, so as to get characters in uniform size standardization is connected. Each fragmented character is standardized in to framework like 32x32 or 64x64. In this way, that all characters have same size.

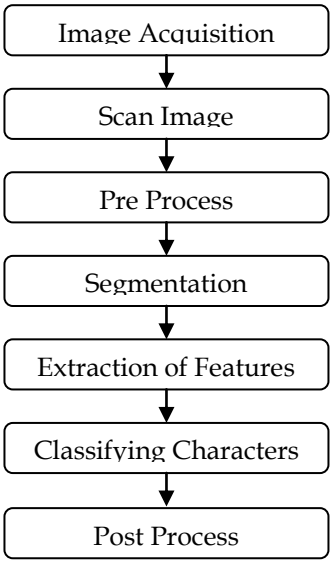


Fig 2: Block Diagram of OCR

Clamor disposal: all in all checked picture may have commotion in it. Commotion in picture is a noteworthy issue in character recognition. Because of the nearness of clamor in the picture, which corrupts the nature of picture, influences on the precision in recognition of picture. Many separating methods are utilized to expel clamor from picture. Clamor disposal is likewise called as smoothing. Decrease of clamor from the picture improves the nature of picture. Clamor in the picture incorporates bending, hole in the lines, inadequate corners and so forth.

Diminishing: The procedure includes expulsion of chose frontal area pixels from parallel picture.

Segmentation: Segmentation is the way toward parceling a picture/record into disjoint and homogenous areas. Division is one of the most significant and fundamental procedure that that improve the precision pace of character recognition framework. Devanagari report is apportioned into grouping of lines and words by vertical and even projection separately [6].

Feature extraction: Feature extraction is characterized as separating the most helpful data from the crude information, which limits the class design fluctuation while upgrading the between class design inconstancy [10]. Feature extraction is significant stage in recognition procedure and furthermore alluded as heart of OCR framework. Highlight extraction procedure extricates the most significant and important shape data present in character [9]. Feature extraction is the uncommon type of Reduction. It diminishes the information when info calculation is extremely huge [8]. Highlight extraction techniques are comprehensively delegated Global Transformation and Series Expansion (for example Fourier Transforms, Gabor Transform and Wavelets), Statistical Features(i.e. zoning, Projection) and Geometrical and Topological Features (for example Separating and Counting Topological Structures and Coding Graphs and Trees and so on) [10]. The strategies like histogram of individual characters and GLCM (Gray level co-event lattice) are likewise considered in highlight extraction for character recognition [7].
Classification: This stage is the basic leadership venture in the optical character recognition framework. There are a few Classical and delicate processing methods accessible for penmanship recognition. Following are the old style methods utilized for characterization.

a) Template coordinating: This is one of the fundamental strategy recognition. Coordinating is utilized to decide the comparability between two points, bends, or states of a similar sort. In format coordinating, a 2D shape or a model of the example to be perceived is accessible.
b) Statistical procedures: In this factual methodology, each example is spoken to regarding d highlights or estimations and is seen as a point in a d-dimensional space.
c) Syntactic Approach: In this methodology, a formal similarity is drawn between the structure of examples and the sentence structure of a language. This methodology considers examples are seen as sentences having a place with a language and natives are seen as the letter set and the sentences are created by a syntax.

Post Processing: Post-handling stage is the last phase of the proposed recognition framework. It prints the comparing perceived characters in the organized content structure.

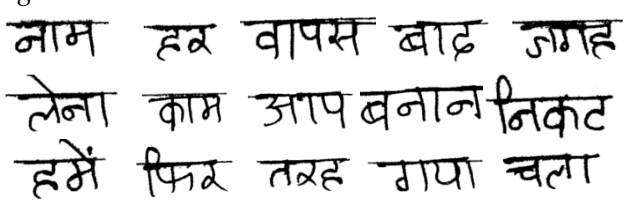
8 Related Study

Keysers et al. [1] used a combination of state-of-the-art components and combine them with novel additions in a flexible framework. This architecture allowed to easily transfer improvements between languages and scripts. Sahare et al. [2] presented robust algorithms for character segmentation and recognition for multilingual Indian document images of Latin and Devanagari scripts. In this, character segmentation algorithm, primary segmentation paths are obtained using structural property of characters, whereas overlapped and joined characters are separated using graph distance theory. Finally, segmentation results are validated using highly accurate support vector machine classifier. For the proposed character recognition algorithm, three new geometrical shape-based features are computed. First and second features are formed with respect to the center pixel of character, whereas neighborhood information of text pixels is used for the calculation of third feature. For recognizing the input character, k-Nearest Neighbor classifier is used, as it has intrinsically zero training time. Vijayalakshmi et al. [3] proposed a diagonal feature-based handwritten character recognition, using multi-layered feed forward neural network with back-propagation algorithm. The pixel information along the two diagonals is primarily extracted and used for recognition. The system is trained and tested on handwritten digits, English lower case character set and Tamil character database and their classification accuracy is compared. S. Ahlawata et al. [4] used hybrid feature set, which consists of multiple feature extraction approaches like Box Method, Mean, Standard Deviation and Centre of Gravity, for recognizing the handwritten numerals. A

Neural network has been used for successfully classifying 550 samples taken from "The Chars74" handwritten numerals dataset. P. P. Bairagiet. al. [5] proposed a new method for recognition of printed Hindi characters in Devanagari script. In this work, different pre-processing methods like features extraction, segmentations and classification have been studied and implemented in order to design a sophisticated OCR system for Hindi based on Devanagari script. M. Yadav et al. [6] focused on detailed survey of existing techniques for recognition of offline handwritten Hindi characters.D. Kumar et al. [7] intended to serve as a guide for the readers, working in the field of off-line handwritten Devanagari character recognition. As of late different techniques have been represented by the researchers in the direction of off-line handwritten Devanagari script recognition. M. Yadav [8] explained available databases for offline Hindi isolated characters and words. It also introduced its readers with the grayscale database proposed in this work. It is the first RGB (colored) database for Hindi handwritten characters. R.K. Maurya [9] reviewed the state of various scripts like Devanagari has attained good results and still improving the accuracy levels. The paper also highlights on the characteristics and challenges of recognition of scripts of Indic origin. Manjunath Aradhya et al. [10] first used PCA for feature extraction and generalized regression neural network (GRNN) was used for efficient classification. This method was tested on printed and handwritten characters of Kannada and English text. The recognition rate achieved was 91%. Sandhya Arora et al. [11] presented an OCR for Handwritten Devanagari Character Recognition by combining multiple feature extraction techniques and used neural as a classifier. On experimentation with a dataset of 4900 samples the overall recognition rate came was 92.80%. Kailash S. Sharma [12] proposed a system which can recognize an online handwritten Devanagari character by self organizing maps. The characters were written with the help of tablet pen. The character written on canvas is then cropped out, converted into fixed size, sampled horizontally and vertically and is mapped on 5x7 matrix template. This matrix is then converted into a vector with thirty five elements which are used as inputs to the two layer Self Organizing Maps network. The number of characters to be recognized are forty. The initial accuracy (recognition rate) of 82% percent can be increased up to 99% on an average, in particular for some user it reaches to 100% by the process of adaptation. J. Pradeep et al. [13] proposed a Neural Network based Handwritten Character Recognition system which was without feature extraction for classifying and recognizing the 26 English alphabets. The neural network architecture with two hidden layers, both having 100 neurons has been found to yield the highest recognition accuracy of 90.19%. Ms. Neha Sahu [14] proposed an efficient handwritten Devanagari character recognition system using artificial neural network technique for recognising Devanagari characters. A small set of Devanagari characters using back propagation neural network was trained, then testing was performed on other word set results in an accuracy of 75.6 % on noisy characters. K. V. Kale [15] et al. proposed a handwritten Devanagari compound character recognition using Legendre moment and neural network was used for recognition The proposed system is trained and tested on 27000 handwritten collected from different people. The overall recognition rate for basic was up to 98.25% and for all compound character was 98.36%. GauriKatiyar1 et al. [16] proposed a hybrid recognition system for off‐line handwritten characters in which multiple features extracted were combined using seven different approaches for better accuracy and reduced computational time for recognition of handwritten characters using Genetic Algorithm which optimized the number of features along with a simple and adaptive Multilayer Perceptron classifier. An accuracy of 91.56 and 87.49 % respectively for capital alphabet and small alphabet respectively was achieved. TalhaIbn Aziz et. al. [17] proposed a directional pattern approach for feature extraction of Bangla numeric characters Local Directional Pattern (LDP) and Gradient Directional Pattern (GDP) were used for feature extraction and then two well-known machine learning algorithms, K-Nearest Neighbour (KNN) and Support Vector Machine (SVM) to classify the numeric character. LDP gives an accuracy of 93.42% and GDP gives a better accuracy of 94.05%. When GDP and LDP were combined the accuracy increases to 94.43%. Vishal A. Naik et al. [18] presented a system to recognize online handwritten character for the Gujarati language. Support Vector Machine (SVM) with linear, polynomial & RBF kernel, k-Nearest Neighbor (k-NN) with different values of k and multi-layer perceptron (MLP) were used to classify strokes using hybrid feature set. This system was trained using a dataset of 3000 samples and tested by 100 different writers. The highest accuracy of 91.63% with SVM-RBF kernel and lowest accuracy of 86.72% with MLP has been achieved. Samuel Talleux et. al. [19] proposed a new approach for handwritten character recognition, using steerable filters and neural networks.Based on a training set of 400 and a set of 50 unknown handwritten digits, 90% correct with less than 1 % wrong recognition was provided. Using only steerable filters, the recognition rate obtained was 74% and it was 72% when only low-pass filtering was used. Moreover a 92% recognition rate can be reached with 60 inputs. N Prameela1 et al. [20] proposed an OCR system for Telugu documents which comprises of three stages, namely pre-processing, feature extraction, and classification. The mean Euclidean distance as well as the mean angular values of the zones were calculated. Lastly SVM classifier was implemented followed by QDA classifier. A comparison analyses among these two was made and listed 80.6% as the recognition rate for SVM and 87.6% for the QDA classifier.

9 Proposed Work

First of all, same data sets are used for implementation by using existing work techniques i.e. SVM along with k-NN. The accuracy with this method is comparatively low as compared to hybrid technique of neural network along with k-NN. The proposed character segmentation algorithm consists of pre-processing, segmentation, post-processing and post verification using SVM to validate the results after segmentation. Data set:



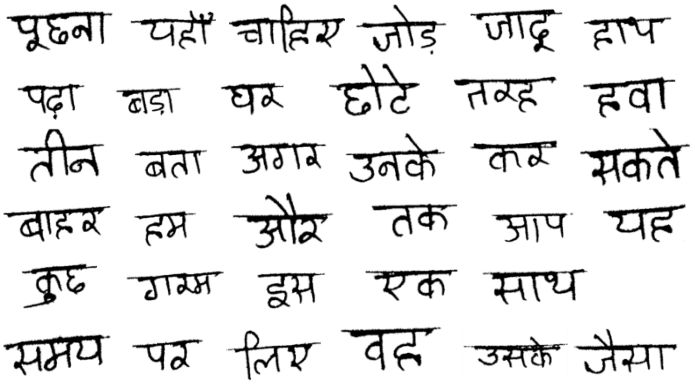


Fig 3: Dataset for Devanagri script words

These steps are described as follows:

1) Pre-Processing: In pre-processing, grey scale word image I of size X × Y having pixel intensity f (m, n) of the pixel located at (m, n) is binarized. Binarization is performed to reduce computational complexity of the algorithm, as only two colors are present for processing. Morphological erosion operation is then performed on the binarized word image to join disconnected components. These components are generally disconnected due to noise. Later, skew of word images is corrected and thinning operation is applied.

2) Character Segmentation: Initially, vertical projection profile of thinned word image is calculated. Let, this thinned image is denoted by Ith of size X × Y having pixel intensity $fth\,(m, n)$ and its vertical projection profile is given as,

$$hn = SumX_{m=1} fth\,(m, n)$$

Here, hn is the number of vertical projection lines varying from 1 to Y. To remove noise and some false projection lines, this projection profile function $h\,n$ is smoothed using Gaussian low pass filter. Now, to find primarily Segmentation Path (SP), only those projection lines are retained [4], which contain values in the range '0' to '2' (as threshold). When these projection lines are mapped on word image, over-segmentation occurs. This over- segmentation is emerged due to the connections or ligatures exist between characters and within the characters like 'r', 'm', 'n', 'u'. To reduce this over segmentation, some of the projection lines are reduced by applying the following conditions. (i) If difference of two projection lines is less than or equal to threshold of 3 pixels, retain the right one and remove the left one. (ii) If difference of two projection profile lines is greater than this threshold, an average of two projection lines is calculated. (iii) Still, if problem of over- segmentation persists and whenever difference of two projection lines is greater than threshold, retain the left one.

3) Post-Processing: Prior to this step, characters that are vertically aligned are segmented accurately. Some joined characters whose ligature contains pixels less than threshold are also segmented in precise manner. However, overlapped and joined characters whose ligatures contain pixels more than threshold are segmented using graph distance theory. For this, grey scale part of two joined characters is considered and their middle column is determined. Here, assumption is made that characters are of equal width and segmentation pixels are high intensity pixels. Now, starting in downward direction and in each iteration, present pixel becomes the center pixel of a searching window 3 × 3, which is shown in Fig 4. As search is in the downward direction, only next row pixels are considered to decrease the redundancy and to make searching mechanism fast and simple. Search for the next pixel is performed using following conditions.

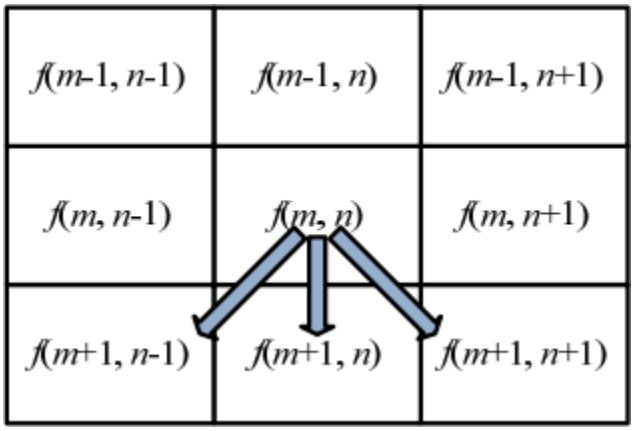


Fig 4: Searching mechanism for finding next segmentation pixel

(i) If present pixel's intensity $f\ (m,\ n)$ = 255, then pixel at location $(m,\ n)$ becomes a segmentation pixel and searching mechanism will start exactly from next row and same column pixel, i.e. from the pixel at location $(m + 1,\ n)$.

(ii) If present pixel's intensity $f\ (m,\ n)$ <= 255, then pixel at location $(m,\ n)$ becomes a segmentation pixel and searching mechanism will start from next row and any of the next column pixels, i.e. from the pixels at locations $(m\ +\ 1,\ n - 1)$, $(m + 1,\ n)$ or $(m\ +\ 1,\ n\ +\ 1)$. Here, these pixel positions are considered as nodes of a graph. Therefore, Euclidean distance weights from graph theory are considered for assigning priorities of next row pixels.

4) Post-Verification: Promising results are obtained for character segmentation ahead of this step. However, over-segmentation problem is still persists in open characters like 'n', 'v', 'm', 'w', 'u'. Therefore, to avoid this problem and validate the segmentation results, post-verification is performed using k-NN classifier. Suppose a training set of label pairs (xi, yi), i = 1, 2, ... m is given, where $xi \in Rn$ and$yi \in \{1, -1\}m$.

Post Verification in Proposed algorithm

Let (Xi, Ci) where i=1, 2……., n be data points.

Xi denotes feature values &Ci denotes labels for Xi for each i.

Assuming the number of classes as 'c'

$Ci \in \{1, 2, 3, \quad , c\}$ for all values of i.

Let x be a point for which label is not known, and we would like to find the label class using k-nearest neighbor algorithms.

Proposed Algorithm:

1) Calculate"$d(x, xi)$" i = 1, 2, … . . , n; where d denotes the Euclidean distance between the points.

2) Arrange the calculated n Euclidean distances in non-decreasing order.

3) Let k be a +ve integer, take the first k distances from this sorted list.

4) Find those k-points corresponding to these k- distances.

5) Let ki denotes the number of points belonging to the

ith class among k points i.e. $k \geq 0$

6) If $ki > kj \forall\ i \neq j$ then put x in class i.

10 Performance Analysis

During pre-preparing, size of structure component for disintegration activity is kept 3 × 3 and limit for Devanagari content is set to 5. While post-handling, Euclidean separation metric is utilized in light of the fact that it speaks to genuine separation between two hubs. In post-check step, SVM with direct bit is utilized, as it separates right and erroneous segments effectively and straight portion is less unpredictable. Criteria to assess execution of proposed character division calculation are Segmentation Rate (SR), Over-Segmentation Rate (OSR) and Bad Segmentation Rate (BSR). These are characterized as pursues:

SR = (Correctly sectioned characters/Total number of characters)*100

OSR = (Over-sectioned characters/Total number of characters)*100

BSR = (Error sectioned characters/Total number of characters)*100

BSR = (Number of accurately ordered examples/Total number of characters)*100

Thus the proposed calculation, if a character is sectioned more than its genuine limits, it is known as over-division. On the off chance that SP does not separate two characters splendidly, it is named as awful division. Rather, SP incorporates some piece of adjoining character while division and makes off base divided characters. Proposed highlights can recognize two comparative showing up characters to a degree. As, characters contain fundamentally various separations among content and focus pixels alongside various number of neighbours. What's more, proposed FCCF is hearty to character scaling since number of cuts in every half stay same regardless of whether character measurements are differed. Four runs are chosen since it is seen that expanding pursues four does not give much improvement in precision. For 10-crease cross approval recognition precision, database is isolated into ten equivalent parts. One of its parts is considered for testing and staying nine subsets are considered for preparing. The dispersion of preparing and testing pictures from various databases is kept in the proportion of 90:10. For CPAR database, out of 83,300 character pictures, 74,970 pictures are utilized for preparing and 8,330 pictures are utilized for testing. It is seen that, 2.7-12.4% higher RRs are gotten on numeral databases, which is because of the nearness of less number of covering classes.

11 Results

The Classification techniques can be categorized as:

a) Classical techniques.
b) Soft computing techniques.

The various classical techniques are Template matching, Statistical techniques, Structural techniques. Whereas the various soft computing techniques are Neural networks, Fuzzy logic, Evolutionary computing techniques.

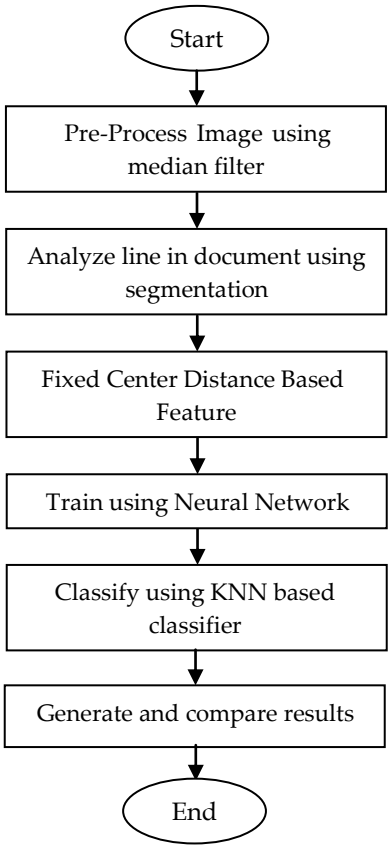


Fig 5: Flow Chart for Research Proposal

Adnan Amin and W. H. Wilson [6] used Neural network for classification of characters with three layers namely Input layer, Output layer and Hidden layer. The geometric features extracted like dot, line, curve or loops are given as input to the input layer. Each component of the segmented representation is classified as dot, line, curve, or loop. In each case, the characteristics of the component are determined: if a line, what are its orientation and its size relative to the character frame - short, medium or long. One input neuron is used to encode each of these possible choices (short/medium/long) and each of four possible orientations for a line. One input neuron is used to encode the characteristics of each component extracted by geometric feature extraction technique.

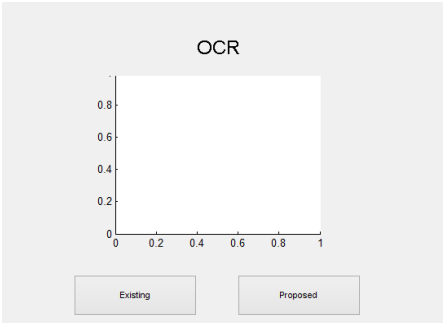


Fig 6: GUI for research work

Figure 6 is the presentation of the GUI developed for the research project. In this research an image is to be uploaded in which that image will be converted into gray scale image and then segmented it. The input images may include the noisy and distorted images also.

Fig 7: Load Input Image

In the figure 7 the image is uploaded in the project. After the uploading of the source image, image will be converted into rgb image so that further operations may be implemented on this image and results may be generated.

Fig 8: Resultant Output Image

Figure 8 is resultant image which is produced with proposed technique.

TABLE 1: PROPOSED CONFUSION MATRIX

n = 50	Predicted No	Predicted Yes	Total
Actual No	TN = 1	FP = 1	2
Actual Yes	FN = 1	TP = 47	48
	2	48	

n = Number of Images
TN = True negative
TP = True positive
FN = False negative
FP = False positive

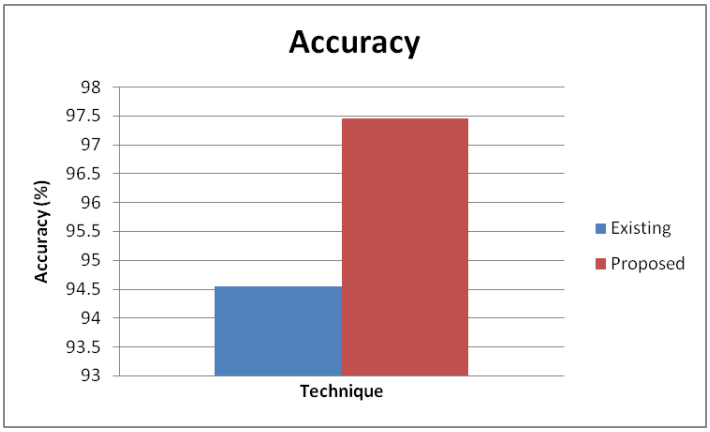


Fig 9: Comparison of existing and proposed methods on the basis of execution time (ms)

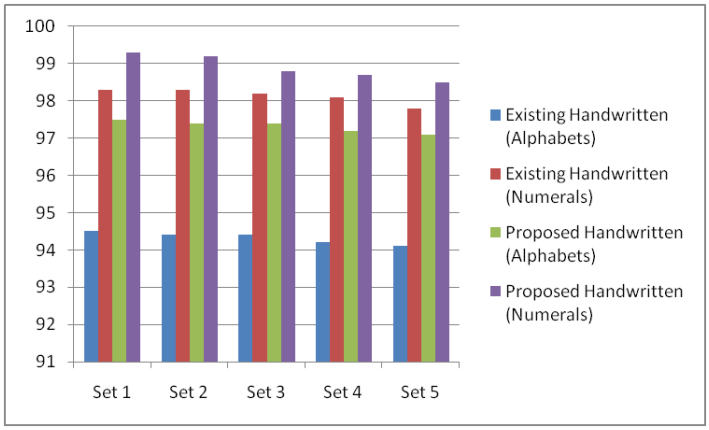


Fig 10: Comparison of existing and proposed methods on the basis of accuracy

The Fig. 10 shows that the accuracy of the proposed technique is more efficient than the existing method.

TABLE 2: ACCURACY COMPARISON OF EXISTING AND PROPOSED APPROACH

Dataset	Type	Set 1	Set 2	Set 3	Set 4	Set 5
Existing	Handwritten (Alphabets)	94.5	94.4	94.4	94.2	94.1
Existing	Handwritten (Numerals)	98.3	98.3	98.2	98.1	97.8
Proposed	Handwritten (Alphabets)	97.5	97.4	97.4	97.2	97.1
Proposed	Handwritten (Numerals)	99.3	99.2	98.8	98.7	98.5

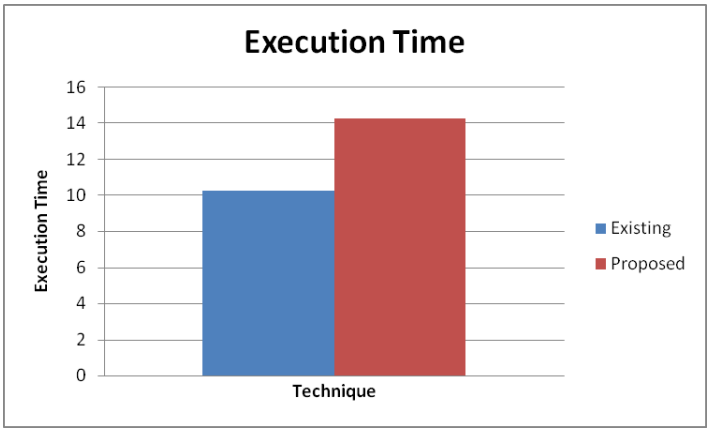


Fig 11: Comparison of Accuracy Existing and proposed approach

TABLE 3: PRECISION COMPARISON OF EXISTING AND PROPOSED APPROACH

Dataset	Type	Set 1	Set 2	Set 3	Set 4	Set 5
Existing	Handwritten (Alphabets)	92.5	92.4	91.4	94.6	92.2
Existing	Handwritten (Numerals)	95.3	94.7	95.5	96.6	94.3
Proposed	Handwritten (Alphabets)	94.6	95.4	96.4	95.2	95.1
Proposed	Handwritten (Numerals)	97.3	97.2	97.8	96.3	98.4

TABLE 4: RECALL COMPARISON OF EXISTING AND PROPOSED APPROACH

Dataset	Type	Set 1	Set 2	Set 3	Set 4	Set 5
Existing	Handwritten (Alphabets)	91.5	92.2	97.4	95.6	92.2
Existing	Handwritten (Numerals)	96.3	94.8	98.5	97.6	94.3
Proposed	Handwritten (Alphabets)	93.6	95.3	97.4	97.2	95.1
Proposed	Handwritten (Numerals)	96.3	97.6	98.8	98.3	98.4

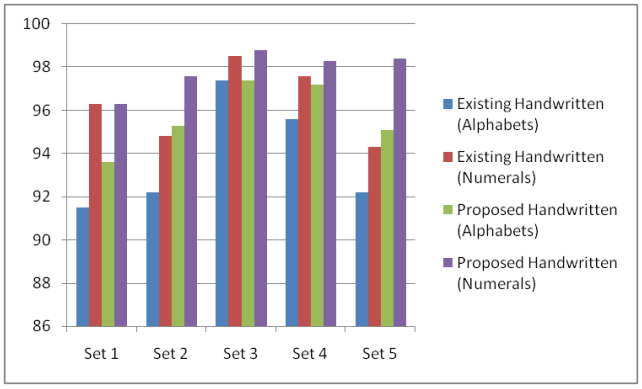


Fig 12: Comparison of Recall Existing and proposed approach

TABLE 5: COMPARATIVE STUDY OF EXISTING AND PROPOSED APPROACH

N = 50	Existing			Proposed		
	Accuracy	Precision	Recall	Accuracy	Precision	Recall
1	92.3	91.0	92.1	94.5	93.2	92.1
2	93.4	92.4	91.2	95.4	93.5	92.4
3	95.4	91.2	93.2	97.4	93.2	94.5
4	94.3	94.2	93.4	96.3	95.4	94.3
5	95.6	92.3	94.2	97.4	93.4	95.6
6	92.6	91.2	94.1	94.8	93.1	92.1
7	92.4	92.4	94.2	95.4	93.2	95.4
8	94.7	92.2	93.2	97.4	95.2	93.5
9	92.3	93.5	92.4	96.3	94.4	94.2
10	93.4	92.3	91.2	97.4	93.4	92.6
11	92.3	91.0	92.1	94.5	93.2	92.1
12	91.4	92.4	91.2	95.4	93.5	92.4
13	92.4	91.2	93.2	97.4	93.2	94.5
14	94.3	94.2	93.4	96.3	95.4	94.3
15	93.6	92.3	94.2	97.4	93.4	95.6
16	91.6	91.2	94.1	94.8	93.1	92.1
17	94.4	92.4	94.2	95.4	93.2	95.4
18	93.7	92.2	93.2	97.4	95.2	93.5
19	92.3	93.5	92.4	96.3	94.4	94.2
20	91.2	92.3	91.2	97.4	93.4	92.6
21	92.3	91.0	92.1	94.5	93.2	92.1
22	93.4	92.4	91.2	95.4	93.5	92.4
23	95.4	91.2	93.2	97.4	93.2	94.5
24	94.3	94.2	93.4	96.3	95.4	94.3
25	95.6	92.3	94.2	97.4	93.4	95.6
26	92.6	91.2	94.1	94.8	93.1	92.1
27	92.4	92.4	94.2	95.4	93.2	95.4
28	94.7	92.2	93.2	97.4	95.2	93.5
29	92.3	93.5	92.4	96.3	94.4	94.2
30	93.4	92.3	91.2	97.4	93.4	92.6
31	92.3	91.0	92.1	94.5	93.2	92.1
32	91.4	92.4	91.2	95.4	93.5	92.4
33	92.4	91.2	93.2	97.4	93.2	94.5
34	94.3	94.2	93.4	96.3	95.4	94.3
35	93.6	92.3	94.2	97.4	93.4	95.6
36	91.6	91.2	94.1	94.8	93.1	92.1
37	94.4	92.4	94.2	95.4	93.2	95.4
38	93.7	92.2	93.2	97.4	95.2	93.5
39	92.3	93.5	92.4	96.3	94.4	94.2
40	91.2	92.3	91.2	97.4	93.4	92.6
41	92.3	91.0	92.1	94.5	93.2	92.1
42	93.4	92.4	91.2	95.4	93.5	92.4
43	95.4	91.2	93.2	97.4	93.2	94.5
44	94.3	94.2	93.4	96.3	95.4	94.3
45	95.6	92.3	94.2	97.4	93.4	95.6
46	92.6	91.2	94.1	94.8	93.1	92.1
47	92.4	92.4	94.2	95.4	93.2	95.4
48	94.7	92.2	93.2	97.4	95.2	93.5
49	92.3	93.5	92.4	96.3	94.4	94.2
50	93.4	92.3	91.2	97.4	93.4	92.6
Mean	93.272	92.27	92.95	96.23	93.8	93.67

10 CONCLUSION

In the available literature, much work is done in recognition of handwritten English letters and various modifications are done it. But, there is much less work done for recognition of Devanagari script. The proposed work to recognize handwritten Devanagari script with more accuracy, precision and recall capacity. The various methodologies utilized in the field of manually written character recognition during the most recent decade have been investigated in the proposed work. Distinctive pre-preparing, division, extraction of features, grouping procedures are additionally examined. However, different techniques for treating the issue of manually written English letters have been created, still a lot of research is required with the goal that a suitable programming arrangement can be made accessible. In this research paper an improved approach for OCR for Devanagari is generated which in turns gave an improved accuracy of 97.4% as compared to 94.5% in existing.

Future Scope: The present diminishing or skeleton discovering calculation relies upon the size of the picture,

which is certainly not a generally excellent methodology towards building up a software like OCR. So an attempt can be made to overcome this circumstance by improving the present diminishing calculation which needs additional time and efforts. More complex data sets can be taken for future and instead of words, line recognition algorithm can be developed.

REFERENCES

[1] D. Keysers, T. Deselaers, H.A Rowley, L.L. Wang, V.Carbune, "Multi-Language Online Handwriting Recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume: 39, No: 6, 2017, page: 1180-1194.

[2] P. Sahare, and S .B. Dhok, "Multilingual Character Segmentation and Recognition Schemes for Indian Document Images", IEEE, ISSN: 2169-3536, Volume: 6, 2018, page: 10603-10617.

[3] K. Vijayalakshmi, S. Aparna, G. Gopal, W. J. Hans, "Handwritten Character Recognition Using Diagonal-Based Feature Extraction", IEEE, 2017, page: 1178-1181.

[4] S. Ahlawata, and R. Rishi, "Off-line Handwritten Numeral Recognition using Hybrid Feature Set – A Comparative Analysis", Information Technology and Quantitative Management, Volume: 122, page: 1092-1099.

[5] P. P. Bairagi, "Optical Character Recognition for Hindi", International Research Journal of Engineering and Technology, Volume: 5, Issue: 5, 2018, page: 3968-3973.

[6] M. Yadav, R. K. Purwar, M. Mittal, "Handwritten Hindi character recognition: a review", IET Image Processing, Volume: 12, Issue: 11, 2018, page: 1919-1933.

[7] D. Kumar, D. Gupta, "Review On Optical Character Recognition for Off-line Devanagari Handwritten Characters & Challenges", International Journal of Scientific Research in Computer Science, Engineering and Information Technology, ISSN: 2456-3307, Volume: 3, Issue: 3, page: 1364-1373.

[8] M. Yadav, "A New Database for Hindi Handwritten Characters", International Journal of Innovations &Advancement in Computer Science, ISSN: 2347–8616, Volume: 7, Issue: 4, 2018, page: 305-309.

[9] R. K. Maurya, "Review of Advances in Digital Recognition of Indic Scripts", International Journal of Scientific Research in Computer Science, Engineering and Information Technology, ISSN: 2456-3307, Volume: 3, Issue: 1, 2018, page: 991-1008.

[10] V. N. Manjunath Aradhya, G. Hemantha Kumar, "Principal Component Analysis and Generalized Regression Neural Networks for Efficient Character Recognition", © 2008 IEEE DOI 10.1109/ICETET.2008.214.

[11] Sandhya Arora, Meghnad Saha, Dipak Kumar Basu, Mahantapas Kundu, "Combining Multiple Feature Extraction Techniques for Handwritten DevanagariCharacter Recognition", 978-1-4244-2806-9/08/$25.00© 2008 IEEE.

[12] Kailash S. Sharma, A. R. Karwankar, Dr. A.S.Bhalchandra, "Devanagari Character Recognition Using Self Organizing Maps", 978-1-4244-7770 2010 IEEE.

[13] J.Pradeep, E.Srinivasan, S.Himavathi, "Neural Network based Handwritten Character Recognition system without feature extraction", 978-1-4244-9394 ©2011 IEEE.

[14] Ms. Neha Sahu, Mr. Nitin Kali Raman, "An Efficient Handwritten Devanagari Character Recognition System Using Neural Network", 978-1-4673-5090 ©2013 IEEE.

[15] K. V. Kale, M. M. Kaz, "Handwritten Devanagari Compound Character Recognition using Legendre Moment an Artificial Neural Network Approach", 978-0-7695-5066-4/13 $26.00 © 2013 IEEE DOI 10.1109/ISCBI.2013.62.

[16] Gauri Katiyar, Shabana Mehfuz, "A hybrid recognition system for off-line handwritten characters", Katiyar and Mehfuz SpringerPlus (2016) 5:357 DOI 10.1186/s40064-016-1775-7.

[17] Talha Ibn Aziz, Al Shahriar Rubel, Md Sirajus Salekin, Rafsanjany Kusho, "Bangla Handwritten Numeral Character Recognition Using Directional Pattern", 978-1-53861150 2017 IEEE.

[18] Vishal A. Naik, Apurva A. Desai, "online handwritten gujarati character recognition using SVM, MLP, AND K-NN", IEEE – 40222.

[19] Samuel Talleux, Vedat Tavsanoglu, Emir Tufan, "Handwritten character recognition using steerable filters and neural networks", 0-7803-4455-3 1998 IEEE.

[20] N. Prameela, P. Anjusha, "Off-line Telugu Handwritten Characters Recognition using optical character recognition", 978-1-5090-5686-©2017 IEEE.