Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.98549369

A COMPREHENSIVE SURVEY ON AUTOMATIC KANNADA HANDWRITTEN TEXT RECOGNITION

Goutam.R
Assistant Professor, Department of Computer Science and Engineering, Atria Institute of Technology, Bangalore-560024, Karnataka, India
goutamsachi@gmail.com

Dr.Vijaya Shetty S
Professor and Head, Department of Computer Science and Engineering, Nitte Meenakshi Institute of Technology, Yelahanka, Bangalore 560064
vijayashetty.s@nmit.ac.in

Abstract— In recent years, researchers have paid more and more attention to Indian handwritten text recognition. India is home to over 1600 distinct languages with 22 recognized as official. In the state of Karnataka, Kannada is the official language. When it comes to the number of native speakers in the world, Kannada comes in at number 33. Numerous scientists have worked to perfect the process of automating optical character recognition. However, the process of handwritten character recognition (HCR) has yet to be enhanced. This survey provides comprehensive guidance for automatic Kannada HCR and demonstrates the significant work needed to create a full HCR system. In this regard, the current survey study sheds light on the advancement of the appropriate techniques to fulfil the goal of building an automatic HCR. The study focuses on all of the steps needed in recognizing Kannada handwriting. The first step is to gather Kannada information that has been written by hand. The next step is to use various processing techniques such as binarization, normalization, noise rejection, morphological approaches, augmentation, picture enhancement, and skew correction to get rid of the stray data points. Character segmentation from Handwritten Kannada text is the third Step. Line, block-wise, and character-based segmentation are all part of the segmentation process. The next step is feature extraction, which takes the segmented character and extracts the salient features. The last step is to use Machine learning (ML) and deep learning (DL) for automatic character recognition from Kannada handwritten script. The purpose of this survey is to help emerging researchers by providing a narrative and detailed analysis of automatic Kannada Handwritten script recognition methods.
Keywords— Kannada, Character Recognition, Handwritten, Feature Extraction, Line Segmentation, Machine Learning, Accuracy.

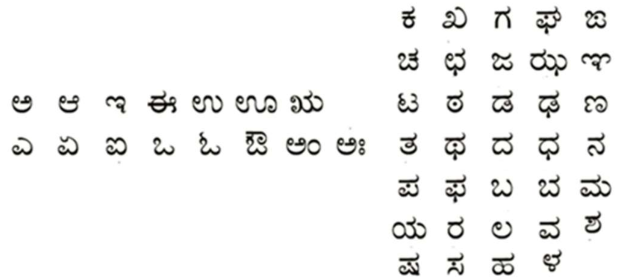
Introduction

The ability to automatically recognize printed text has come a long way in recent years. But there is no technology available that offers accurate recognition of handwritten text. Handwritten recognition has long been a focus of document image analysis due to its wide range of possible uses. Potential uses for text comprehension include data entry, form filling, as well as cheque processing, and the assessment of answer scripts [1]. The flexibility of character design—its ability to be stretched, swooped, tilted, stylized, crushed, connected, tiny, massive, etc.—presents a significant challenge to the development of a reliable identification system. Manually transcribing large amounts of handwritten data is a laborious and error-prone operation. Reduce the time spent on the transformation of high-level data to text and lay the groundwork for new uses with automated handwritten recognition. There has been a significant increase in research on the use of AI, computer vision, and pattern recognition to decipher handwritten writing. Handwriting recognition systems can be either online or offline. Both kinds can be used to implement the applications incrementally in response to user feedback. Several methods, both online and offline, have been created for handwritten recognition [2]. Recognizing text has evolved into a difficult problem. The vast variety of individual handwriting styles and the generally lower quality of handwritten text compared to printed language are two major factors contributing to the difficulty of converting and understanding the handwritten text. All industries, from banking and insurance to healthcare, must work together to solve this urgent problem [3]. Reliable and effective pre-processing and segmentation are required to achieve the desired result in the word recognition stage. Better OCR performance is impossible without proper segmentation. We need segmentation to create an effective OCR for word recognition. Because of the structural intricacy of the Kannada language, handwritten text is more challenging to segment than text written in Latin-based languages [4]. This is because Kannada has vowels and consonants and looks to have a mixture of overlapping Ottaksharas and Vyanjanas. This laborious process has been happening for many years. The segmentation of handwritten Kannada writing into lines, words, and characters is crucial for some specialized applications due to the language's large character set. Because there are more modifier characters in the handwritten Kannada script, segmenting it is difficult.

Reading aids for the blind, reading numbers on bank checks, automatically interpreting pin codes to sort mail, and so on are just a few of the many real-world applications that have helped boost interest in HCR studies [5]. The identification of written Indian characters has been the subject of extensive study. The recognition of handwritten characters, however, has seen comparatively few attempts. For scripts like Devanagari, Bangla, and Kannada, the vast majority of research in this area has been on the recognition of handwritten characters in an offline setting. More than sixty journals dealing with automatic character recognition using image processing, ML, and DL were examined. Search terms like "Kannada," "handwritten," "segmentation," and "artificial intelligence" were used to pull articles from Scopus, IEEE, Science Direct, and Google Scholar.

Kannada Language Description

Roughly 48 million people can communicate fluently in Kannada, the official language of Karnataka. Kannada's language and script are heavily influenced by Bramhi. During the fifth and seventh century AD, the Brahmi-related Kadamba and Chalaukya scripts are the ancestors of the Kannada alphabet. When compared to the Latin alphabet, the Kannada script is very different. Unlike the characters of many north Indian languages, Kannada characters don't feature shirorekha (a line that unites all the characters in a word), therefore they all appear independently. Word segmentation is made challenging as a result. Because of its complex characters, the Kannada script is more challenging to learn than English. The upper- and lower-case characters of this script are not distinguished [6]. Kannada characters (vowels and consonants) are illustrated in Figure 1 in alphabetical order [8].



(a) Vowels (b) Consonants

Fig 1. Kannada characters

The 52 characters of the Kannada script are made up of 16 vowels and 36 consonants. There are 10 different decimal-based Kannada numerals included in each writing system. The fundamental consonants and vowels can also be modified with alternative symbols. The numerical value of these augmentations is identical to that of the original characters. Known as aksharas, these characters are made by combining consonants, consonant modifiers, and vowel modifiers by established rules of combination. Because there are 16 vowels and 36 consonants, there are 576 consonant-vowel combinations and 20736 consonant-consonant-vowel combinations [7].

Difficulties with handwritten Kannada [9]:

- Strokes and handwritten styles differ drastically from one another.
- Because of unpredictable handwriting, the font style of a handwritten document may change from time to time.
- Smudges of ink appear on the paper as time passes.
- The poor quality of the source material makes identification difficult.
- Separation is tough with cursive handwriting recognition, and analysis takes time.
- Printed writing is usually easy to interpret and recognize in a system where all the characters sit up straight, whereas handwritten text can be invariant and have a different rotation.

Process Flow

For automatic recognition of Kannada handwritten text, the system should follow five stages which are illustrated in Figure 2. The purpose and working of each stage are detailed below.

- Data: Data collection is the initial step for automatic text recognition. If the data is more, the recognition rate will be accurate.
- Pre-process: The collected handwritten data contain noise, blur, and much garbage. To clean the raw data, some pre-processing steps are involved.
- Segmentation: Segmentation is important to extract characters from the handwritten script.
- Feature Extraction: The important features are retrieved from the segmented characters.
- Recognition: Using the extracted features, the Kannada text is recognized using ML and DL approaches.

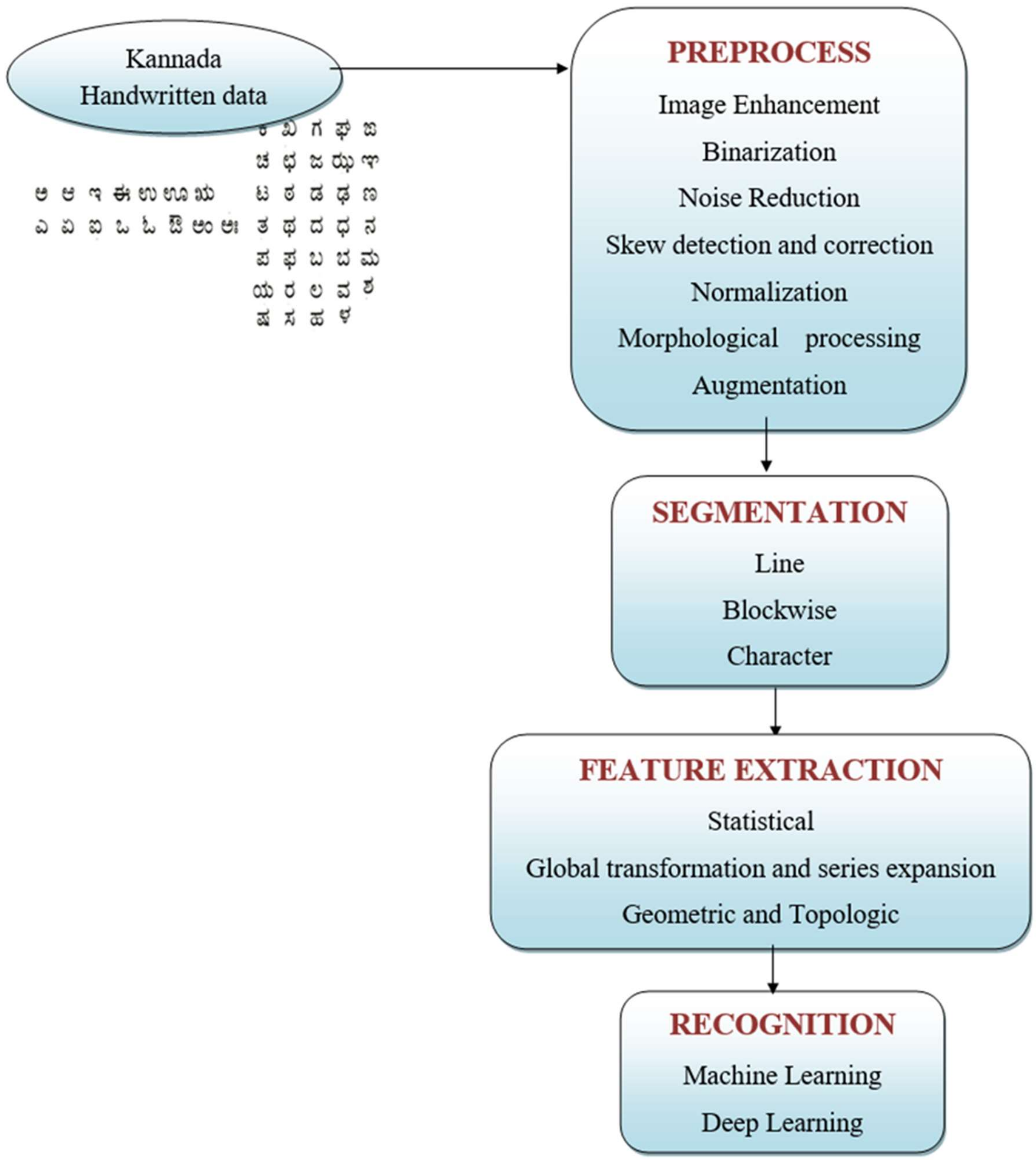


Fig. 2. Flow of Kannada handwritten text recognition using AI

Data Acquisition

There is no consistent dataset for handwritten and printed Kannada alphabets and numerals, as far as we are aware. As a result, we assemble the handwritten archives of writers from various backgrounds to generate a dataset of entirely unfettered Kannada alphabet and numerals. The following is a description of available Kannada handwritten characters datasets.

Most of the study made use of the Chars74K dataset [10]. There are 16 vowel characters and 34 consonant characters in the Kannada alphabet, which has 50 classes. Images of handwritten characters illustrate each class. The data set includes handwritten numbers 0-9 in Kannada. The collection contains images of handwritten characters in various handwriting styles.

On the Kaggle website [11], there is a large image collection of scanned Kannada handwritten characters. Each of the collection's 657 categories comprises 25 images, for a total of 16,425 images. The linguistic data collection consists of vowels (swaras), consonants (vyanjana), and gunitakshara.

Image Processing

The handwritten data has undergone pre-processing, a series of steps designed to eliminate redundancies and make the text legible. Noise must be removed from the initial greyscale image, the background must be smoothed, and contrast must be increased, all of which can only be accomplished through pre-processing. The processing methods utilized for Kannada handwritten recognition are described in greater detail in [12].

Image enhancement:

Reducing or eliminating distracting aspects including noise, blurring, lack of contrast, and lack of information is central to image enhancement's mission [13]. Making an image better than the original is the primary goal of image enhancement. It is possible to enhance the visual quality of images in several different ways. The final imaging task has a significant effect on the approach selection. Enhancing an image means making it more legible, both to humans and to automatic image processing technologies. Spatial domain image-enhancing techniques include thresholding and mask processing [14].

Binarization

Through a process called "image binarization," a picture is simplified into a black-and-white format where white and black pixels indicate the background and foreground [15]. Preprocessing is only possible for grayscale or binary images, where the pixel density value is between 0 and 255 in the former case and between 0 and 1 in the latter, where 0 indicates white and 1 indicates black, respectively. In the process of "binarization," a grayscale image is transformed into a binary one via a thresholding technique. By choosing a thresholding value, images in grayscale or color can be represented as binary images.

Noise Reduction:

Removing meaningless bit patterns from the output is the primary goal of noise reduction. Noise can be reduced by a variety of techniques, including filtering, morphological processes, and modeling. Among the many possible filter designs are those that smooth, sharpen, threshold, remove subtle background texture, and alter contrast. Characters can be thinned out, boundaries can be removed, and disconnected strokes can be joined using bespoke morphological techniques. Filters are a great tool for achieving this smoothing effect [16]. The following explains the types of filters [17,18].

- Linear filter: Modifications such as low pass filtering, sharpening, and smoothing can be applied. Grayscale image noise was greatly diminished by using a low pass filter, including the Gaussian or average blur filter.
- Non-Linear Filters: When it comes to eliminating noise in individual pixels of grayscale images, the median filter came out on top. Median filters take an image and replace the center pixel with the median value by first selecting a section of the image, then sorting through all of the pixel values within that region. If there are an even number of pixels in

the region of interest, the middle two will be used as a mean. Impulse noise is made up of random occurrences of black and white pixels, and the median filter successfully eliminates this type of noise. Using a median filter, we can filter out the unwanted background noise in the image. Compared to convolution, a median filter is superior at decreasing noise while maintaining crisp edges.

Skew Detection and Correction:

When the baseline of some text is not quite horizontal, we say that the text is skew. If feed pages into a scanner at an angle, the resulting scan will be crooked. Skew detection influences page segmentation and classification. Skew detection is necessary for text or document image alignment so that text lines are parallel to coordinate axes [19]. There are several different types of skew detection techniques.

- One way to find parts that fit together is to measure the normal angle between their centers.
- Variations in the number of black pixels per projected line can be used as a proxy for the projection angle of a given document page, as demonstrated by a profile analysis of projections.
- If we project a ray through the image in a parallel fashion, some of the rays will go through blank areas (between lines of text), while others will pass through many letters quickly and then hit several dark areas.

Once the crooked angle has been identified, the page can be rotated accordingly. To fix this distortion, a rotational transformation is applied. Changing the below equation is as simple as changing the coordinate axes.

$$\left(x' \; y'\right) = \left(cos\theta \; sin\theta \; - sin\theta \; cos\theta\right)\left(x \; y\right) \qquad [1]$$

Normalization:

It is a linear process. If the image's intensity range is 50 to 180 but like it to be 0 to 255, can do this by subtracting 50 from the intensity of each pixel, giving a range of 0 to 130. To create a scale from 0 to 255, we multiply the intensity of each pixel by 255/130. Auto-normalization is often achieved by utilising the full arithmetic range of the image file format. Following normalization [20], all of the generated zones will have the same fixed dimensions. Standardized data are obtained by normalization operations with the intent of eliminating all writing-related variations [21]. To make text appear uniform in size, for instance, size normalization can be utilized. Horizontal and vertical size normalizations are possible in character recognition techniques.

To find the boundaries of an image without taking into account random spots of noise, researchers have devised a Neighbourhood Search-based method. W_1 and H_1 represent the width and height of a character, W_2 and H_2 represent the n normalized width and height of a character, and L stands for the standard plane size. Common dimensions for a flat surface is 32× 32 or 64×64 pixels. The unaltered and adjusted aspect ratios, R_1 and R_2, are as follows:

$$R_1 = \frac{min(W_1,H_1)}{max\,(W_1,H_1)} \qquad [2]$$

$$R_2 = \frac{min(W_2,H_2)}{max\,(W_2,H_2)} \qquad [3]$$

Morphological Processing:

The morphological processing involved in character recognition are detailed [22] below.

- Thinning: This morphological approach, like erosion or opening, can be applied to binary images to remove carefully selected foreground pixels. Thinning is a data reduction technique in which an item is reduced down to one pixel in width, creating a skeleton of the thing that makes letter recognition easier.
- Erosion: Erosion shrinks an item by chipping away at the pixels that define its edges.
- Dilation: The dilation technique enlarges an object by adding border pixels.
- Closing: It's similar to a dilatation, but not quite. The image's foreground tends to get larger. It is less damaging to the original border form. Closing restores or replaces missing sections and fills unattractive gaps.
- Opening: It's very similar to erosion. It tends to remove some foreground pixels at the edges of foreground regions. It is not as destructive as typical erosion. Things that are too close can be separated, things that should not be touching can be disconnected, and holes within objects can be widened by opening them.

.

Augmentation

This method is used to artificially increase a dataset when working with minimal image data. This is especially useful when there are just a few data samples available for each image class. Because of a shortage of data, overfitting is a risk in DL. We also had a limited number of images to deal with in each categorization, thus image augmentation was used to supplement our data [23]. This is accomplished through the usage of the ImageDataGenerator function.

Segmentation

Segmentation is the practice of extracting characters from an image that are of interest for further study. The process of isolating each character from a handwritten image is known as segmentation [24]. During segmentation, the image is broken down into its constituent pieces. Segmentation is the method of splitting an image into multiple portions that convey different meanings. Local and global segmentation are the two most used approaches to the segmentation of images [25]. In local segmentation, a mask is created by first segmenting smaller portions of an image. In contrast to global segmentation, local segmentation works with a significantly smaller number of pixels. In terms of pixel requirements, local segmentation needs to be creative. "Global segmentation" means to divide an entire image into smaller parts. Projected parameter values are more trustworthy for global segmentation since the method works best with segments having a high number of pixels. Three primary paradigms can be applied when segmenting an image: the region technique, the boundary method, and the edge approach. Line segmentation, block-wise segmentation, character segmentation, and many other methods exist for segmenting images of old Kannada handwritten texts [26].

Line Segmentation

There are still a lot of national archives and manuscript libraries that haven't been digitized and made available to the public. Instead of reading entire pages automatically, we employ longterm objective tasks including authentication, word detection, text/image alignment, and field extraction. The first step in any one of these processes is to divide the document into individual lines of text. Automatic text line segmentation is not well-studied because of the poor quality and complexity of these documents (noise, age-related aberrations, uneven illumination, interfering lines) [27]. Compared to modern books, ancient manuscripts, whether printed or handwritten, have much more flexible layout formatting standards. This makes it more challenging to obtain their physical structure. The absence of detail in the typeface, along with the natural decay of old papers, makes them illegible. The holes, decorations, inverted lettering, stains, and seals are particularly unsettling [28]. Sometimes, when someone is writing, the lines on the page will get too close together, if not touch. The author, period, and environment all play a role in giving words and characters their idiosyncratic and unique twists. The vocabulary is extensive, and it even includes some obscure phrases and proper nouns. However, specialized OCR can be made for printed documents, so at least that use case is covered for the time being.

Projection, stochastic, grouping, Repulsive-Attractive network, Hough transform, smearing, processing of overlapping and touching components, and many other methods have all been utilized in the literature to isolate text lines from the document image.

Block-wise Segmentation

Block-wise segmentation [29] is a technique that divides a digital document image into smaller regions (blocks) that each contains only one type of information (text, halftone image, graphic, etc.). Some white pixels in the intermediate bitmap created by block-wise segmentation have been converted to black to achieve a specific adjacency of patterns related to the same type of data. The block-wise segmentation technique has 4 stages.

Character recognition in strings
Using the length-height matrix to classify
Page divide based on borders
Block Unionisation

A block-wise segmentation strategy was determined to be the optimum way to divide document images into text and figure sections [30]. To begin, character string collections are connected. We can determine the merger threshold by evaluating the height and distance histograms of character strings. The margins of the text groups are then linked to determine the column bounds. The margins are then used to divide the rest of the page into sections. Cohesion is then created by utilizing the spatial relationships between the elements on the page. These merged blocks are then separated into distinct sections for the text and figures using a projection profile.

Character Segmentation

Due to their status as the fundamental building blocks of every written language, character segmentation is a vital part of any character recognition system [31,32]. The method can convert a scanned or printed document's image into a text file that can be read. The segmentation algorithm has a significant impact on the recognition system's accuracy. To correctly identify characters, clean character segmentation is essential. Words are difficult to split into characters in handwriting because the script is fluid and unstructured.

There has been a long history of study into the problems of character segmentation and recognition. Even now, it presents difficulties in image processing and pattern detection. The job of segmentation is to locate appropriate letter boundaries. Segmentation occurs before character recognition; therefore its results are fed into the processing pipeline of the latter. The method for character segmentation relies on a linked component analysis of the foreground components (objects) of the binary image, regardless of whether the text was printed by machine or typed by hand.

Feature Extraction

As feature extraction is a critical step in character recognition, the majority of research has focused on developing measures that express a pattern succinctly while still including enough information to allow reliable recognition. Feature extraction is the process of extracting information from raw data that is most relevant for categorization to maximize variability across classes while decreasing variability within them [33]. In general, ideal traits will include the following: A modest within-class variation is required. The second requirement is a significant disparity between social groupings. For the character to be recognized as having multiple versions, certain features must be invariant under specific transformations [34]. We discuss the various feature extraction algorithms used for character recognition.

Statistical

From the distribution pattern of the points, we can infer these features. They are simple to use, yield fast results, and are flexible enough to accommodate a range of users' formatting preferences. The feature set can be narrowed down with their help. The most important statistical features are as follows:

- Zoning: The features are gleaned by a close examination of the densities of the points and specific strokes in a variety of segmented, overlapping, or non-overlapping zones within the character's frame. It is possible to evaluate a character's contour directions using their contour direction attributes [35]. Points of flexion provide yet another illustration. The places in an image where a stroke curves sharply are called bending points.
- Loci: Horizontal and vertical vectors are generated for each white point in the character's background, and the frequency with which these vectors intersect line segments is a feature.
- Distance and Crossings: Crossing is a measure of how many pixels in the character's image transition from the background to the foreground along horizontal and vertical lines [36]. Distances are measured horizontally from the first detected pixel to the image's top and bottom edges.

Series Expansion and Global Transformation

These properties are unaffected by global transformations such as translations and rotations. When identifying a continuous signal, there is generally more data to represent. A signal can be expressed as a sequence of linear combinations of well-defined and smaller functions. The linear combination coefficients are used in sequence expansion, a space-saving encoding [37]. Transforms and series expansions have the following properties:

- Fourier Transform (FT): In most cases, features in n-dimensional Euclidean space are chosen based on the magnitude spectrum of the measurement vector. One of the FT's most enticing properties is its ability to recognize position-shifted characters when only the magnitude spectrum is analysed. FTs have been used in numerous OCR systems.
- Hough Transform: It's a method for determining document baselines. It's also used to define character parameter curves [38].
- Walsh Hadamard Transform (WHT): Because the only arithmetic computations required by this feature are addition and subtraction, it lends itself well to high-speed processing [39]. The fundamental issue with this transformation is that it is extremely sensitive to character placement.
- Rapid transform: Because it is identical to the WHT except for the absolute value function, the position-shifting issue may have been resolved.
- Karhunen Loeve expansion: It is a type of Eigenvector analysis in which linear combinations of the original features are used to generate new features with lower dimensionality [40].
- Gabor Transform: This method is similar to the windowed FT. In this case, rather than a fixed value, a Gaussian function is used to determine the window size [41].
- Wavelets: The wavelet transform is a series expansion technique that produces a variety of signal representations with varied resolutions.
- Moments: The objective of moment normalization is to remove the translation and rotation limitations from object detection in images of various sizes.

Geometrical and Topological Features

Characters' global and local attributes can be expressed through these features, and they are also quite forgiving of stylistic differences. These topological features may require familiarity with the object's various component kinds or encode information about its shape [42].

- Strokes: All characters start with these traits. It's possible to use everything from simple straight lines and curved circles to more complicated curves and splines to create Kannada letters [43]. For optical character recognition, a stroke is characterized as the vertical distance between the bottom and top of the pen.
- Bays and Stroke Directions: How a pen moves across the paper to create a character's features.
- Coded chains: The effect is achieved by mapping the strokes of a character into the parameter space of a computer program [44].
- The intersections of straight and curved lines and loops.
- Angle properties and interactions between strokes.

Classification

In computer science, "artificial intelligence" is shorthand for a broad category of methods that draw from many fields to boost hardware and software efficiency. ML is a branch of AI concerned with finding solutions to difficult problems by the automatic, unsupervised discovery of relationships between variables. The goal of ML techniques is to create accurate predictions with minimal reliance on inference, as opposed to the purpose of traditional statistical models, which are hypothesis-driven and attempt to uncover links between outcomes and data points. The "deep learning" discipline of ML employs a deep artificial neural network architecture to discover hidden patterns in data.

Input data is classified by the recognition algorithms, which are then evaluated and refined with both trained and unseen data. Character classification is a frequent application of ML. Examples of supervised methods include the K-Nearest Neighbour [45], Naïve Bayes [46], Support vector Machine [47], decision tree, random forest [48], and Linear Discriminant Analysis [49], whereas unsupervised methods include the FCM, self-organization map, and SSAE.

The findings suggest that DL is a well-liked method for handwritten application recognition. Using tagged image datasets and the SoftMax activation function in the output layer, supervised training is the most common method for DL-based image classification. This method determines which class best reflects the object in the image. As new Convolutional Neural Network (CNN) architectures have been developed and put into practice, image classification has gained traction among researchers across disciplines. AlexNet [50], LeNet [51], VGGNet [52], ZFNet, GoogLeNet, ResNet [53], CapsNet [54], DenseNet [55], and SENet are some of the most recent advances in CNN designs for character recognition. The comparison of recent works on Kannada HCR using AI is given in Table 1.

Table 1. Analysis of Kannada handwritten character recognition using AI

Model	Data	Pre-process	Accuracy
Hybrid Feature Extraction with Ensemble DL (HFE-EDH) [56]	Own	Image Enhancement, Noise Reduction	96%
CNN [57]	Chars74k	Gray Conversion, Binarization, Noise Reduction, Normalization	98%
CNN [58]	Chars74k	Gray Conversion, Resize, Image Enhancement, Binarization, Noise Reduction, Normalization	96.7%
CapsNet [59]	Own	Not Mentioned	99%
CNN-SVM [60]	Own	Binarization, Noise Reduction, Morphological Operation	98%
HCR-Net [61]	Kannada-Mnist	Augmentation	98.27%
Morlet Stacked Sparse Auto Encoder [62]	Chars74K	Binarization	96.73%
CNN-XGBoost [63]	Own	Not Mentioned	99.63%
CNN-RNN [64]	Char74K	Gray Conversion, Resize,	86.82%

		Noise Reduction, Normalization	
Two-stage multi modal DL [65]	Kaggle	Not Mentioned	89%

Conclusion

HCR is a fascinating and difficult subfield in pattern recognition. Kannada handwritten records remained the sole means of documentation found in government organizations and hospitals in Karnataka; replicating the contents of these old papers via typewriting was going to be a time-consuming, labor-intensive operation that could result in errors while documentation. As a result, the government may use the automatic HCR model to reproduce historical documents written in Kannada. In this study, we conducted a literature review of ML and DL approaches to Kannada HCR. Handwritten Kannada character identification is an extremely demanding problem to solve. From data collection to recognition, this survey includes summaries of all the steps involved in Kannada HCR. Analysis of the existing literature reveals that an excellent character recognition rate requires a careful selection of categorization and feature extraction approaches. The bulk of researchers are concentrating on the recognition of Kannada handwritten characters. Yet, there is no universal technique for accurately identifying all of the characters used in Kannada. At each stage of the recognition process, many methods have been employed, but ultimately, each method is capable of offering a solution for a limited range of Kannada characters. There are still obstacles to be overcome in the recognition process, such as the inability to distinguish between normal and unusual writing, slanted characters, similarly shaped characters, arcs, and so on.

References

- De Campos, T. E., B. R. Babu, and M. Varma. "Character recognition in natural images Proceedings of the International Conference on Computer Vision Theory and Applications (VISAPP)." Lisbon, Portugal, February (2009).
- Mamatha, H. R., and K. Srikantamurthy. "Morphological operations and projection profiles based segmentation of handwritten Kannada document." International Journal of Applied Information Systems (IJAIS) 4, no. 5 (2012): 13-19.
- G.Azhaguvarthani, B. Ramani “Handwritten text recognition system for English” international journal of pure and applied mathematics, 117 , no. 16 (2017): 257-263,.
- Sheshadri, Karthik, Pavan Kumar T. Ambekar, Deeksha Padma Prasad, and Ramakanth P. Kumar. "An OCR system for Printed Kannada using k-means clustering." In 2010 IEEE international conference on industrial technology, pp. 183-187. IEEE, 2010.
- Ouali, Imene, Mohamed Saifeddine Hadj Sassi, Mohamed Ben Halima, and W. A. L. I. Ali. "A new architecture based ar for detection and recognition of objects and text to enhance navigation of visually impaired people." Procedia Computer Science 176 (2020): 602-611.
- Kumar, Munish, Manish Kumar Jindal, Rajendra Kumar Sharma, and Simpel Rani Jindal. "Character and numeral recognition for non-Indic and Indic scripts: a survey." Artificial Intelligence Review 52 (2019): 2235-2261.

- Indira, K., and S. Sethu Selvi. "Kannada character recognition system a review." arXiv preprint arXiv:1001.5352 (2010).
- Rajput, Ganpat Singh G., and Rajeshwari Horakeri. "Zone based handwritten Kannada character recognition using crack code and SVM." In 2013 international conference on advances in computing, communications and informatics (ICACCI), pp. 1817-1821. IEEE, 2013.
- Sampathkumar, S. "Kannada Handwritten Answer Recognition using Machine Learning Approach." (2022).
- http://www.ee.surrey.ac.uk/CVSSP/demos/chars74k/
- https://www.kaggle.com/datasets/dhruvildave/kannada-characters?select=Img
- Alginahi, Yasser. "Preprocessing techniques in character recognition." Character recognition 1 (2010): 1-19.
- Devi, A. Saradha, S. Suja Priyadharsini, and S. Athinarayanan. "A block based scheme for enhancing low luminated images." arXiv preprint arXiv:1009.1478 (2010).
- Maini, Raman, and Himanshu Aggarwal. "A comprehensive review of image enhancement techniques." arXiv preprint arXiv:1003.4053 (2010).
- Mustafa, Wan Azani, and Mohamed Mydin M. Abdul Kader. "Binarization of document images: A comprehensive review." In Journal of Physics: Conference Series, vol. 1019, no. 1, p. 012023. IOP Publishing, 2018.
- Mostafavi, S. Mohammad, Iman Abaspur Kazerouni, and Javad Haddadnia. "Noise removal from printed text and handwriting images using coordinate logic filters." In 2010 International Conference on Computer Applications and Industrial Electronics, pp. 160-164. IEEE, 2010.
- Win, Ni Ni, Khin Khat Khat Kyaw, Thu Zar Win, and Phyo Phyo Aung. "Image noise reduction using linear and nonlinear filtering techniques." Int. J. Sci. Res. Publications 9, no. 8 (2019): 816-821.
- Hambal, Abdalla Mohamed, Zhijun Pei, and Faustini Libent Ishabailu. "Image noise reduction and filtering techniques." International Journal of Science and Research (IJSR) 6, no. 3 (2017): 2033-2038.
- Cao, Yang, Shuhua Wang, and Heng Li. "Skew detection and correction in document images based on straight-line fitting." Pattern Recognition Letters 24, no. 12 (2003): 1871-1879.
- Wang, Wen-Li, and Mei-Huei Tang. "A normalization process to standardize handwriting data collected from multiple resources for recognition." Procedia Computer Science 61 (2015): 402-409.
- Konovalenko, Ivan Andreevich, Vladislav Vladimirovich Kokhan, and Dmitry Petrovich Nikolaev. "Optimal affine image normalization approach for optical character recognition." Компьютерная оптика 45, no. 1 (2021): 90-100.
- Alkoffash, Mahmud S., Mohammed J. Bawaneh, Hasan Muaidi, Shihadeh Alqrainy, and Muath Alzghool. "A survey of digital image processing techniques in character recognition." International Journal of Computer Science and Network Security (IJCSNS) 14, no. 3 (2014): 65.

- Shorten, Connor, and Taghi M. Khoshgoftaar. "A survey on image data augmentation for deep learning." Journal of big data 6, no. 1 (2019): 1-48.
- Zheng, Yefeng, Huiping Li, and David Doermann. "The segmentation and identification of handwriting in noisy document images." In Document Analysis Systems V: 5th International Workshop, DAS 2002 Princeton, NJ, USA, August 19–21, 2002 Proceedings 5, pp. 95-105. Springer Berlin Heidelberg, 2002.
- Chaubey, Ashutosh Kumar. "Comparison of the local and global thresholding methods in image segmentation." World Journal of Research and Review 2, no. 1 (2016): 1-4.
- Kaur, Rupinder Pal, M. K. Jindal, and Munish Kumar. "TxtLineSeg: text line segmentation of unconstrained printed text in Devanagari script." In Computational Methods and Data Engineering: Proceedings of ICMDE 2020, Volume 2, pp. 85-100. Singapore: Springer Singapore, 2020.
- Kumar, Munish, Rajendra Kumar Sharma, and Manish Kumar Jindal. "Segmentation of lines and words in handwritten Gurmukhi script documents." In Proceedings of the first international conference on intelligent interactive technologies and multimedia, pp. 25-28. 2010.
- Malgi, PRAMOD S., and S. H. A. I. L. J. A. Gayakwad. "Line segmentation of Devanagari handwritten documents." Int J Electron, Commun Instrum Eng Res Dev (IJECIERD) 4, no. 2 (2014): 25-32.
- Hirayama, Yuki. "A block segmentation method for document images with complicated column structures." In Proceedings of 2nd International Conference on Document Analysis and Recognition (ICDAR'93), pp. 91-94. IEEE, 1993.
- Gllavata, Julinda, Ralph Ewerth, and Bernd Freisleben. "A text detection, localization and segmentation system for OCR in images." In IEEE Sixth International Symposium on Multimedia Software Engineering, pp. 310-317. IEEE, 2004.
- Casey, Richard G., and Eric Lecolinet. "A survey of methods and strategies in character segmentation." IEEE transactions on pattern analysis and machine intelligence 18, no. 7 (1996): 690-706.
- Lu, Yi. "Machine printed character segmentation—; An overview." Pattern recognition 28, no. 1 (1995): 67-80.
- Ramappa, Mamatha Hosalli, and Srikantamurthy Krishnamurthy. "A Comparative Study of Different Feature Extraction and Classification Methods for Recognition of Handwritten Kannada." International Journal of Database Theory and Application 6, no. 4 (2013): 71-90.
- Suganya, Suganya. "Analysis of Feature Extraction of Optical Character detection in Image Processing Systems." International journal of engineering research and technology 3, (2018)
- Herwanto, Heru Wahyu, Anik Nur Handayani, Katya Lindi Chandrika, and Aji Prasetya Wibawa. "Zoning feature extraction for handwritten javanese character recognition." In 2019 International Conference on Electrical, Electronics and Information Engineering (ICEEIE), vol. 6, pp. 264-268. IEEE, 2019.

- Mohamad, Muhammad'Arif, Haswadi Hassan, Dewi Nasien, and Habibollah Haron. "A review on feature extraction and feature selection for handwritten character recognition." International Journal of Advanced Computer Science and Applications 6, no. 2 (2015).
- Siddharth, Kartar Singh, Mahesh Jangid, Renu Dhir, and Rajneesh Rani. "Handwritten Gurmukhi character recognition using statistical and background directional distribution." Int. J. Comput. Sci. Eng.(IJCSE) 3, no. 06 (2011): 2332-2345.
- Touj, Sofien, Najoua Essoukri Ben Amara, and Hamid Amiri. "Generalized Hough Transform for Arabic Printed Optical Character Recognition." Int. Arab J. Inf. Technol. 2, no. 4 (2005): 326-333.
- Shrivastava, Vivek, and Navdeep Sharma. "Artificial neural network based optical character recognition." arXiv preprint arXiv:1211.4385 (2012).
- Grother, Patrick J. "Karhunen Loeve feature extraction for neural handwritten character recognition." In Applications of Artificial Neural Networks III, vol. 1709, pp. 155-166. SPIE, 1992.
- Huang, Yu, and Mei Xie. "A novel character-recognition method based on gabor transform." In Proceedings. 2005 International Conference on Communications, Circuits and Systems, 2005., vol. 2. IEEE, 2005.
- Chaudhuri, Arindam, Krupa Mandaviya, Pratixa Badelia, Soumya K Ghosh, Arindam Chaudhuri, Krupa Mandaviya, Pratixa Badelia, and Soumya K. Ghosh. Optical character recognition systems. Springer International Publishing, 2017.
- Chiu, Hung-Pin, and Din-Chang Tseng. "A novel stroke-based feature extraction for handwritten Chinese character recognition." Pattern Recognition 32, no. 12 (1999): 1947-1959.
- Mohamad, Muhammad Arif, Jamaludin Sallim, and Kohbalan Moorthy. "Whale Optimisation Freeman Chain Code (WO-FCC) Extraction Algorithm for Handwritten Character Recognition." In 2021 International Conference on Software Engineering & Computer Systems and 4th International Conference on Computational Science and Information Management (ICSECS-ICOCSIM), pp. 598-602. IEEE, 2021.
- Dhandra, B. V., Mallikarjun Hangarge, and Gururaj Mukarambi. "Spatial features for handwritten Kannada and English character recognition." Special Issue on RTIPPR-10, International Journal of Computer Applications (2010): 146-150.
- Joe, Kevin George, Meghna Savit, and K. Chandrasekaran. "Offline character recognition on segmented handwritten Kannada characters." In 2019 global conference for advancement in technology (GCAT), pp. 1-5. IEEE, 2019.
- Dutta, Kusumika Krori, Sunny Arokia Swamy, Anushua Banerjee, R. Chandan, and Deepak Vaprani. "Kannada character recognition using multi-class SVM method." In 2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence), pp. 405-409. IEEE, 2021.
- Dutta, Kusumika Krori, Ashita Victor, Archana G. Nathu, Mohammed Ayman Habib, and Devanshi Parashar. "Kannada alphabets recognition using decision tree and random forest models." In 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), pp. 534-541. IEEE, 2020.

- Prasad, M. Mahadeva, and M. Sukumar. "2D-LDA based online handwritten Kannada character recognition." Int. Jl. of Computer Science and Telecommunications 4, no. 1 (2013): 14-18.
- Rani, N. Shobha, N. Chandan, A. Sajan Jain, and H. R. Kiran. "Deformed character recognition using convolutional neural networks." International Journal of Engineering & Technology 7, no. 3 (2018): 1599-1604.
- Prashanth, Duddela Sai, R. Vasanth Kumar Mehta, Kadiyala Ramana, and Vidhyacharan Bhaskar. "Handwritten devanagari character recognition using modified lenet and alexnet convolution neural networks." Wireless Personal Communications 122 (2022): 349-378.
- Rani, N. Shobha, A. C. Subramani, Akshay Kumar, and B. R. Pushpa. "Deep learning network architecture based Kannada handwritten character recognition." In 2020 second international conference on inventive research in computing applications (ICIRCA), pp. 213-220. IEEE, 2020.
- Al-wajih, Ebrahim, Rozaida Ghazali, and Yana Mazwin Mohmad Hassim. "Residual neural network vs local binary convolutional neural networks for bilingual handwritten digit recognition." In Recent Advances on Soft Computing and Data Mining: Proceedings of the Fourth International Conference on Soft Computing and Data Mining (SCDM 2020), Melaka, Malaysia, January 22– 23, 2020, pp. 25-34. Springer International Publishing, 2020.
- UÇAR, Emine, and Murat UÇAR. "Applying capsule network on Kannada-MNIST handwritten digit dataset." Natural and Engineering Sciences 4, no. 3 (2019): 100-106.
- Sandhya, S., and V. Geetha. "Isolated Kannada Character Recognition using Densely Connected Convolutional Network." In 2022 International Conference on Asian Language Processing (IALP), pp. 137-142. IEEE, 2022.
- Parashivamurthy, Supreetha Patel Tiptur, and Sannangi Viswaradhya Rajashekararadhya. "Recognition of Kannada Character Scripts Using Hybrid Feature Extraction and Ensemble Learning Approaches." Cybernetics and Systems (2023): 1-36.
- Asha, K., and H. K. Krishnappa. "Kannada handwritten document recognition using convolutional neural network." In 2018 3rd international conference on computational systems and information technology for sustainable solutions (CSITSS), pp. 299-301. IEEE, 2018.
- Parikshith, H., SM Naga Rajath, D. Shwetha, C. M. Sindhu, and P. Ravi. "Handwritten character recognition of kannada language using convolutional neural networks and transfer learning." In IOP conference series: materials science and engineering, vol. 1110, no. 1, p. 012003. IOP Publishing, 2021.
- Rani, N. Shobha, and Pushpa BR. "Robust recognition technique for handwritten Kannada character recognition using capsule networks." International Journal of Electrical & Computer Engineering (2088-8708) 12, no. 1 (2022).
- Ramesh, G., Sandeep Kumar, and H. N. Champa. "Recognition of Kannada handwritten words using SVM classifier with convolutional neural network." In 2020 IEEE Region 10 Symposium (TENSYMP), pp. 1114-1117. IEEE, 2020.

- Chauhan, Vinod Kumar, Sukhdeep Singh, and Anuj Sharma. "HCR-Net: A deep learning based script independent handwritten character recognition network." arXiv preprint arXiv:2108.06663 (2021).
- Ganeshaiah, Keerthi Prasad, and Vinay Hegde. "Handwritten Character Recognition Using Morlet Stacked Sparse Auto Encoder based Feature Dimension Reduction."
- Ramesh G, P. N. Sharada, B. Padmavathy, “Hybrid Classifier-Based for Offline HandwrittenKannada Digit Recognition,” Neuro Quantology 20, Issue 13, (2022): 2601-2615
- Siddanna, S. R., Anant Kumar, Avdhesh Dubey, Bal Govind, and Diksha Dixit. "Kannada handwritten character recognition using hybrid RNN."
- Siddanna SR, Kiran YC. “Two Stage Multi Modal Deep Learning Kannada Character Recognition Model Adaptive to Discriminative Patterns of Kannada Characters” Indian Journal of Science and Technology. 16, (2023):155-166.