Journal of Ambient Intelligence and Humanized Computing (2024) 15:3275–3287
https://doi.org/10.1007/s12652-024-04805-y

ORIGINAL RESEARCH

A top-down character segmentation approach for Assamese and Telugu handwritten documents

Prarthana Dutta[1,2] · **Naresh Babu Muppalaneni**[1,3]

Received: 10 May 2023 / Accepted: 14 April 2024 / Published online: 7 May 2024

Abstract
Digitization offers a solution to the challenges associated with managing and retrieving paper-based documents. However, these paper-based documents must be converted into a format that digital machines can comprehend, as they primarily understand alphanumeric text. This transformation is achieved through Optical Character Recognition (OCR), a technology that converts scanned image documents into a format that machines can process. A novel top-down character segmentation approach has been proposed in this work, involving multiple stages. Our approach began by isolating lines from handwritten documents and using these lines to segment words and characters. To further enhance the character segmentation, a *Raster Scanning* object detection technique is employed to isolate individual characters within words. Thus, the character segmentation results are integrated from the results of the vertical projection and raster scanning. Recognizing the significance of advancing digitization of handwritten documents, we have chosen to focus on the regional languages of Assam and Andhra Pradesh due to their historical and cultural importance in India's linguistic diversity. So, we have collected datasets of handwritten texts in Assamese and Telugu languages due to their unavailability in the desired form. Our approach achieved an average segmentation accuracy of 93.61%, 85.96%, and 88.74% for lines, words, and characters for both languages. The key motivation behind opting for a top-down approach is two-fold: firstly, it enhances the accuracy of character recognition, and secondly, it holds the potential for future use in language/script identification through the utilization of segmented lines and words.

Keywords Computer vision · Optical character recognition · Character segmentation · Raster scanning

1 Introduction

Despite being a paperless world, some people (and even situations) prefer the traditional mode of writing with pen on paper. In order to store, access, and share such documentation in the future, there arises a need for digitizing all these documents. Thus, *digitization* is a simple process of converting image-based content into machine-readable content. Optical Character Recognition, or OCR, has been a hot research topic over the days of yore for its worldwide application domains. It is the process of transforming or converting handwritten or printed image-based content into an editable format for various purposes such as access, transfer, store, etc. It passes through several phases where every phase has its own significance and importance in recognizing the printed or handwritten text (Singh et al. 2012; Joseph 2022). These phases are preprocessing, segmentation, feature extraction, recognition, and classification.

The OCR is a vast topic of research and is divided into a number of different categories, which are individually explored taking different languages under consideration (Ahamed et al. 2020a; Girdher et al. 2022; Singh et al. 2023; Rahman et al. 2022; Srivastava et al. 2022; Kaur et al. 2022). Basically, it is divided into two categories: *Online and Offline*—depending on the mode of acquisition of the text, and *Handwritten and Printed*—depending on the mode of writing.

The OCR builds itself as an automated advancement toward improving the human–machine interface in many

✉ Prarthana Dutta
prarthana.dutta01@gmail.com

Naresh Babu Muppalaneni
nareshmuppalaneni@gmail.com

[1] Department of Computer Science and Engineering, National Institute of Technology, Silchar, Assam 786003, India

[2] Department of Computer Science and Engineering, The Assam Kaziranga University, Jorhat, Assam 785006, India

[3] Indian Institute of Information Technology Design and Manufacturing Kurnool, Kurnool, Andhra Pradesh, India

ways. It can benefit from advancements in multimedia processing, improving its ability to handle complex documents. A semantic understanding of the OCR processes can improve recognition accuracy, helping to interpret the content of the recognized text. OCR can be integrated into wearable devices to provide instant access to information by recognizing text in the wearer's surroundings. Thus, the recognition of a language, text, word, or character has been emerging as an important and necessary field of research in various application domains such as banking, healthcare systems, multimedia databases, etc. Among many application domains in Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL), handwriting character recognition is one of the active and fascinating areas utilizing image processing and computer vision (Pastor-Pellicer et al. 2016; Renton et al. 2017; Chen et al. 2017; Grüning et al. 2019).

Most of the research on the recognition of text is done for languages and scripts such as Arabic (Ali and Suresha 2020), Chinese (Chen et al. 2021), Roman (Abdulhussain et al. 2021), etc. Also, few attempts have been carried out to recognize Indic scripts such as Devanagari, Bangla, Tamil, Oriya, and Gurmukhi handwritten scripts (Chirimilla and Vardhan 2022). However, several regional languages have not received sufficient attention in this regard. Consequently, the authors have chosen to focus on regional languages due to the limited availability of datasets, literary resources, and research experimentation on these languages in existing literature.

Most of the Indian scripts are constituted of some structural characteristics, which makes them different from other non-Indic scripts. Some of these properties are:

(i) There are no Upper and Lower case letters or characters.
(ii) They have a set of Basic Characters (Vowels and Consonants). The Assamese and Telugu languages have a set of 11 and 16 vowels, respectively, and 41 and 36 consonants, respectively.
(iii) Both languages have their set of compound and composite characters with their unique form and structure in document writing. These types of characters are formed by combining a vowel with a consonant or a consonant with another.
(iv) Constitutes various *Zones* (Upper, Middle, and Lower) and *Lines* (Upper Line, Head Line, Base Line, and Lower Line), shown in Figs. 1 and 2.
(v) Another special feature seen in some languages (such as Assamese, Hindi, etc.) is the component called the "*Matra*"—a horizontal line on the Head line of a character or word (Fig. 1). This component is absent in the Telugu language.

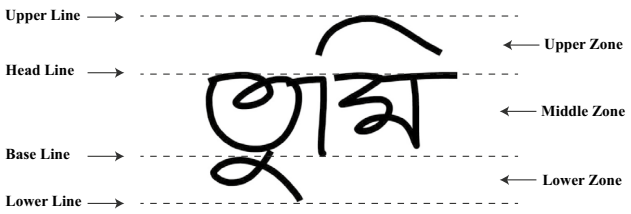

Fig. 1 Various zones and lines of a word in Assamese

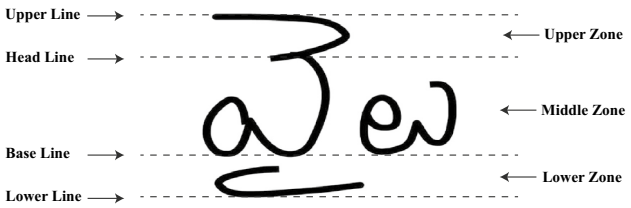

Fig. 2 Various zones and lines of a word in Telugu

(vi) May also contain diacritics and accents.
(vii) In both languages, using spaces helps readers identify individual words and aids in overall readability. Spaces are used to separate words, much like in English. Spaces are placed between words, allowing readers to identify where a word ends and the next begins, thus demarcating word boundaries.

1.1 The Assamese and Telugu language

The Assamese language, also known as Asamiya, is the official language of the Indian state of Assam and is spoken by approximately 20 million of the Indian population. It has its roots in the older Kamrupi language spoken in ancient Kamrup, which corresponds to present-day Assam (Bose 1989). The evolution of the Assamese language was greatly influenced by the works of Srimanta Shankardeva, a revered saint-scholar of Assam, particularly his composition of the Kirtan Ghosha (narratives of Krishna). The script used for writing Assamese is derived from the ancient Brahmi script in 300 BC. During the British colonial era, Assamese gained recognition as the official language of Assam. Over time, Assamese literature has flourished, encompassing a wide range of genres, including spiritual texts, poetry, plays, and more. The language holds deep historical and cultural significance in northeastern India as a key Assamese identity and heritage element. It continues to thrive as an integral part of the region's linguistic and cultural tapestry.

Telugu is the official language of the Indian states of Andhra Pradesh and Telangana and is spoken by 96 million

of the Indian population. It has ancient origins rooted in the Dravidian language family. It boasts a rich literary heritage, with notable contributions from poets, scholars, and a prominent saint-scholar, Bhadra Bhupala, throughout history. The Telugu script, an abugida, is used for writing the language. In the modern era, Telugu is recognized as an official language and is widely spoken in India and by the Telugu diaspora worldwide. Its cultural and linguistic significance remains strong and essential to Telugu identity and heritage.

1.1.1 Challenges in the Assamese and Telugu handwritten text and characters

Some of the challenges in the handwritten texts and characters of both languages are:

(i) Scarcity of complex datasets in the required form for various tasks such as segmentation, recognition, etc.
(ii) Writing style may vary from individual to individual.
(iii) Presence/Absence of the upper headline ("Matra") component.
(iv) Presence of different diacritics.
(v) Presence of slants within the text.
(vi) Variation in the font style and size.
(vii) Variability in stroke thickness and curvatures.
(viii) Presence of Character Ambiguity, may pose a challenge in segmentation and recognition as well.

These challenges make the recognition and segmentation task a challenging one for regional languages like the Assamese and Telugu.

Hence, digitizing these documents (Assamese and Telugu) is urgently needed to preserve this valuable linguistic and cultural heritage and make it accessible to a wider audience (Batchas and Shahid 2021; Krishna and Ram 2021). The main motivation for this work is inspired by the idea of developing an approach for recognizing lines, words, and characters from offline handwritten documents in regional languages.

1.2 Contribution and organization

The main contribution of the present work is credited to the collection of handwritten Assamese and Telugu documents. The novel contribution of our work is character segmentation from the textual documents written, which is addressed in the context of two different languages—Assamese and Telugu. These languages exhibit distinct structural features, character shapes, and stroke patterns despite being regional languages. In Assamese, the "*Matra*" is a unique element in many characters, representing a single or fusion of consecutive characters to form a word. In Fig. 1, the head line that comprises a character or a word forms a matra. Notably, Telugu lacks this Matra component. Our segmentation approach is designed to handle both text types, accommodating those with and without the Matra line and showcasing its versatility in addressing diverse linguistic characteristics and writing styles in these languages.

The successive parts of the paper are organized as follows: literature ideas of the current state of the line, word, and character segmentation works are presented in Sect. 2. The dataset collection and our proposed top-down approach for character segmentation are explained in Sect. 3. Section 4 and Sect. 5 present the experimental results of the three levels of segmentation and discuss the final observations. Finally, Sect. 6 concludes and presents future scopes and directions of research in the present domain.

2 Background

There are a number of segmentation approaches proposed for handwritten datasets, such as the counting pixel approach (Malik et al. 2020), GAN (Kundu et al. 2020), LineCounter (Li et al. 2021), etc. Significant advancements have been made in achieving impressive segmentation and recognition accuracies, accompanied by extensive comparisons of these approaches across diverse datasets and models. The final goal of recognizing characters from handwritten documents passes mostly through three levels (Dutta and Muppalaneni 2022), and the relevant recognition is attained at various levels (Chatterjee et al. 2019). The traditional hierarchy for segmentation initiates from script identification, followed by language identification, which is further followed by page, line, word, and character segmentation. Script identification is one of the crucial recognition phases in literature, and many studies have been witnessed to attain some great satisfactory results (Cheikhrouhou et al. 2021; Singh and Sachan 2020; Ukil et al. 2020; Ahmad et al. 2020b). Language identification is also explored in literature where different languages belonging to the same or different script family are recognized from document texts (Mioulet et al. 2015; Zouari et al. 2019).

A counting pixel-based approach is used to segment the handwritten and printed Urdu document images (Malik et al. 2020). Darker pixel values are considered to be constituent parts of the text which are to be segmented. Hence, the consecutive rows having dark pixels are extracted as lines. The white pixel values are considered as separating sections between two lines. This approach gains 98.1% line recognition accuracy on the Urdu handwritten texts.

Generative Adversarial Networks, or GANs, effectively utilize text line recognition in handwritten documents. Kundu et al. explored the GANs on the ICDAR 2013 and the HIT-MW datasets (Kundu et al. 2020). They utilized

two GAN architectures: Encoder–decoder and the U-Net, which gave pretty good *F* measure scores on both datasets. Projection profiling-based methods have attained heights and are utilized for text line recognition. Adaptive thresholding approached based on the projection profile method is used to isolate handwritten text lines of the Uyghur language (Suleyman et al. 2021).

Another line recognition approach for handwriting documents was given by Li et al. in their work (Li et al. 2021), which counted the number of lines present at each location of the pixel in the document image. They named this line-counting approach the "Linecounter" which determines the line number while traversing the pixels from the top. An efficient method of using horizontal projection and scale-space method for recognizing text lines and words was brought about by Rajyagor and Rakholia (2021).

Dutta et al. (2021) took up working on complex structured handwritten and printed data with distortions by developing a learning-based approach. The approach is based on a text line and background partitioning idea. An Encoder–Decoder-based CNN model is designed to learn features and perform classification efficiently. A similar idea of pixel counting of text lines and background was taken by Barakat et al., where they employed an unsupervised method for text line recognition (Barakat et al. 2021)-testing the approach on challenging datasets such as Arabic (VML-AHTE), ICDAR, etc.

Recognition of handwritten text lines and the constituent words on the Meitei Mayek dataset was performed in Inunganbi et al. (2021). The horizontal and vertical projection profiles played good roles in the recognition of text lines and words, respectively. After correctly recognizing the lines in the dataset document with an accuracy of 91.84%, words are identified with the help of the Vertical Projection Histogram. Words are detected at locations where there are separating points at the start and end of the rows and columns. A word is identified when the vertical projection is 0 for all the columns. In situations where the column is not detected for word identification, four points are detected, two above and two below, which help in word detection. This approach attained a word recognition accuracy of 88.96%.

A simple neural network model called the "Origami-Net" is proposed in (Yousef and Bishop 2020) for recognizing text lines in the IAM and ICDAR2017 datasets. Their proposed mechanism attained a character error rate of 6.5% and 6.8%, respectively. Zhou et al. proposed a hybrid binarization algorithm and an adaptive character extraction approach to segment the characters for serial number recognition from banknotes (Zhou et al. 2019). Their approach proved to segment the character boundaries more precisely by adopting a pre-segmentation ideology.

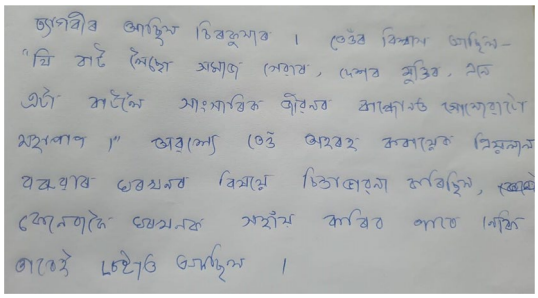

Fig. 3 Sample of the collected Assamese handwritten text document

Here, they adjusted the boundaries based on the spatial distribution of the character strokes.

3 Methodology

In the present work, we collect datasets of Assamese and Telugu languages from native writers. The dataset is passed through the preprocessing and a top-down segmentation phase to segment or isolate text lines, words, and characters.

3.1 Dataset collection

As a crucial phase for any recognition task, the dataset is a challenge in our work. Though various datasets (online, offline, handwritten, and printed) are publicly available for languages such as Arabic, Devanagari, Chinese, Roman, etc., very few are available for regional Indian languages like Assamese and Telugu. To address this, we gathered our dataset by reaching individuals from Assam and Andhra Pradesh, regardless of age, gender, or educational background, to write short texts on plain paper with a pen. Each individual provided handwriting samples of approximately 6–8 lines, which were then captured using mobile cameras and scanned at 300 dpi. We have collected 200 samples of Assamese and 222 samples of Telugu handwritten text documents. We believe this dataset size is sufficient for our study. A few samples of the collected dataset of Assamese and Telugu are shown in Figs. 3 and 4 respectively. Our data is taken on different environmental conditions like the GNHK dataset (Lee et al. 2021) in different light intensities. Though the image is captured with the same mobile specification, the type of paper on which the text is written is not kept the same, which greatly brings some variations in the quality of the input along with the inclusion of noise. The dataset of Assamese

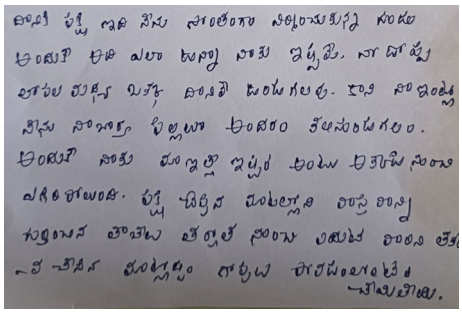

Fig. 4 Sample of the collected Telugu handwritten text document

and Telugu handwritten text is uploaded on the IEEE data port[1] (Dutta and Muppalaneni 2024).

3.2 Pre-processing

After obtaining all the scanned images, it is very important to pass these images through a set of preprocessing to clean them, standardize them, and make the best use of the image information for processing. The pre-processing techniques are employed in such a manner that they can handle noisy data. The various pre-processing of the text documents carried out in our study are:

Grayscale conversion: The three-channel camera captured image is converted into a single-channel image to simplify further processing. The color channel is insignificant in this task, so it is converted to the grayscale with intensity values ranging between 0 to 255.

Binarization: Binarization is converting the grayscale image to the binary (black and white) format of 0 and 1 pixel. Otsu's Binarization is a popular technique employed here (Bangare et al. 2015) that calculates an optimal threshold to separate the foreground (text) from the background. We thus obtain an image with only black and white colors. It helps in enhancing the contrast between text and background.

A Gaussian blur filter is applied to the grayscale image before binarization. Gaussian blur can help reduce noise and smoothen the image to make it easier to determine the optimal threshold using Otsu's method. This efficient and relatively fast filtering technique removes noise and blurs the image. This filter also tends to preserve edges in an image to some extent, thus achieving a balance between noise reduction and retaining important image features.

Deskewing: The skewness of an image is the measure of the extent of slant or rotation of the image from a horizontal or vertical alignment. It can be removed by rotating the image with the same skew amount in the opposite direction. In our proposed approach, for detecting and correcting any skew in the image, we defined three parameters: *delta* (the step size for angle variation), *limit* (range of angles), and *angles* (an array of angles to be tested). The algorithm iterates through the range of angles to find the optimal angle for deskewing. It rotates the image for each angle, calculates a histogram, and computes a score based on the sum of the squared differences between consecutive values in the histogram. The angle that produces the highest score is considered the best angle for deskewing. Once the best angle for deskewing is obtained, the image is straightened by rotating it by this angle in the opposite direction to remove the skew.

Morphological opening: It is applied to remove any imperfection in the image caused by binarization and restore the original structure of the image. A 3×3 square filled with ones is the structuring element (kernel) used in our approach. The structuring element defines the shape and size of the neighborhood used for the morphological operation. This 3×3 structural element for the morphological opening is chosen in our approach to incorporate a balance between preserving relevant information and removing unwanted noise in the image. It treats pixels in all directions equally due to its symmetric structure. While performing the morphological opening, a certain amount of noise removal task is also done. The kernel moves through the binary image, and at each position, it computes the minimum value within its neighborhood. This operation has the effect of "shrinking" or "eroding" the binary image's white (foreground) regions. As a result, small isolated white regions (noise or small unwanted features) smaller than the kernel size tend to get removed or reduced. Thus, the morphological opening operation effectively removes noise in binary images by eroding small isolated white regions in the image, often considered noise.

The raw and the corresponding pre-processed images are shown in Figs. 5 and 6.

3.3 Top-down segmentation

Many researchers follow the traditional top-down means to achieve the character level of segmentation (Obaidullah et al. 2019; Chatterjee et al. 2019; Dutta and Muppalaneni 2022). The top-down approach usually starts from the page levels and ends at attaining the character level of segmentation. This hierarchy can be seen in Fig. 14.

3.3.1 Line segmentation

Isolating the individual lines from a text document is called text-line or simply line segmentation. It is considered an important stage for many OCR systems since inaccurate

[1] https://ieee-dataport.org/documents/assamese-and-telugu-handwritten-text-dataset.

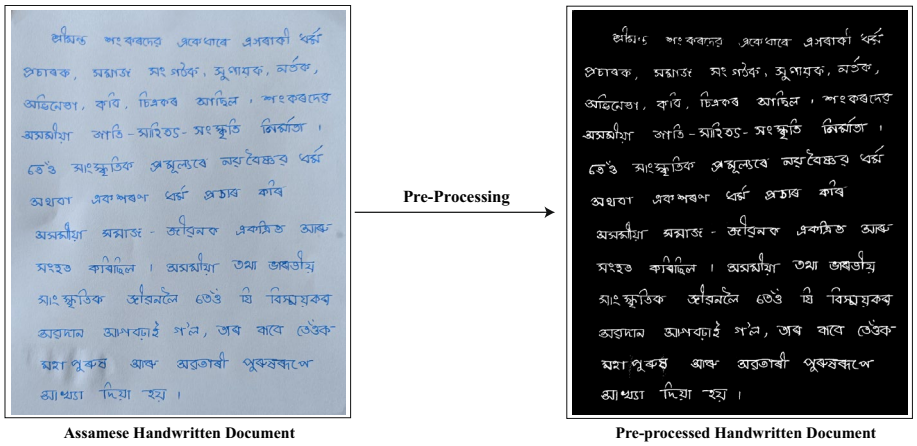

Fig. 5 A raw handwritten Assamese document and the corresponding pre-processed image

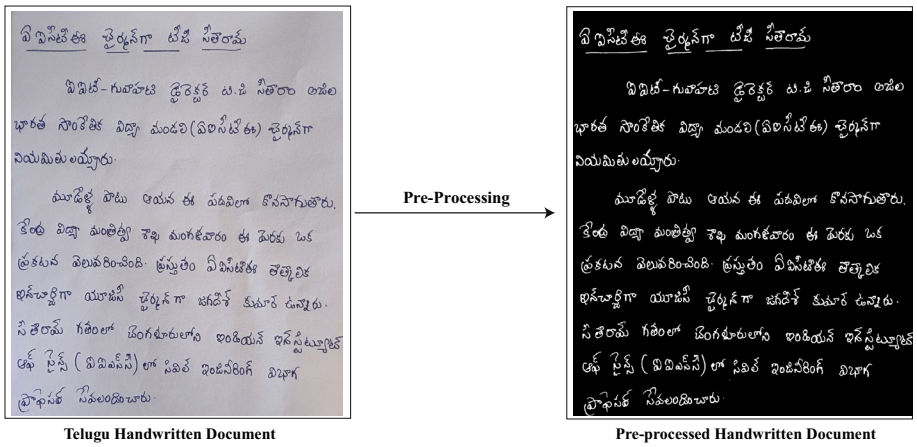

Fig. 6 A raw handwritten Telugu document and the corresponding pre-processed image

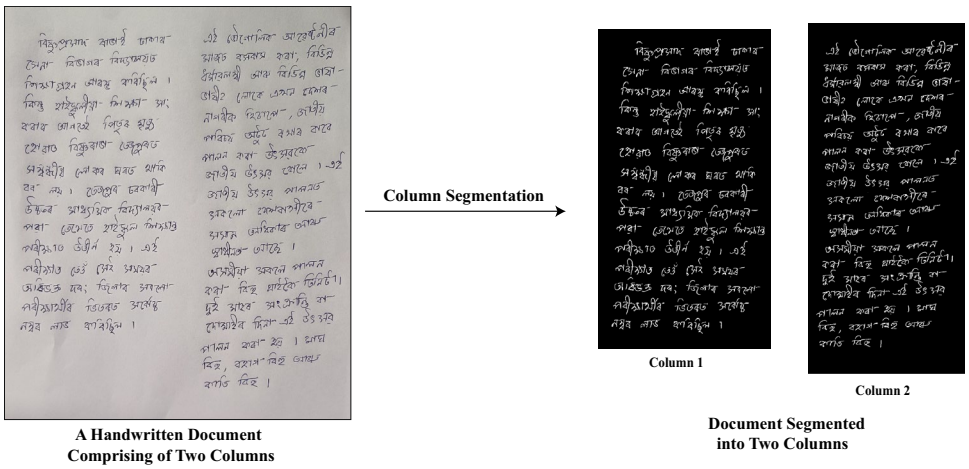

Fig. 7 A sample handwritten document in two columns is separated into constituent columns

recognition may lead to errors in the subsequent stages of recognition.

Before performing line segmentation, it may be necessary to separate multiple columns on a page or document into individual columns. This can be better understood from Fig. 7 and is described in Algorithm 1.

The algorithm initially identifies contiguous regions within the text image, which can either be areas with markings or blank areas. The regions with markings are the potential regions containing the handwritten text, while the blank regions correspond to areas where the vertical projection equals zero. These blank regions, termed separating regions, serve as boundaries between columns in the text. By

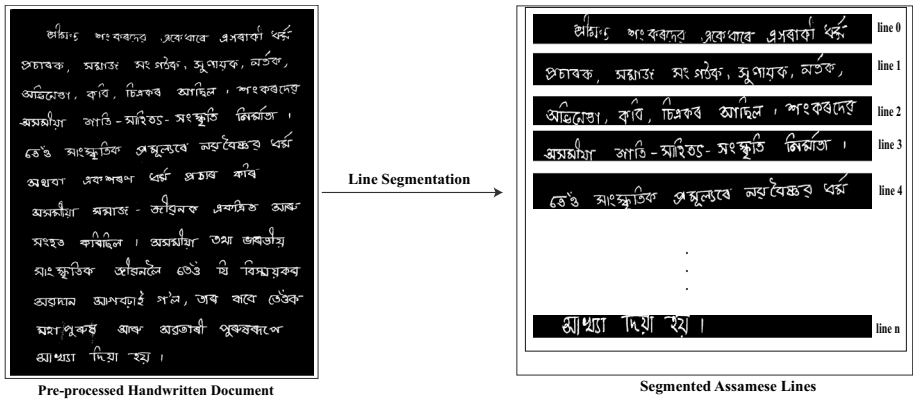

Fig. 8 Pre-processed Assamese document and the corresponding segmented lines obtained after line segmentation

distinguishing between marking and non-marking regions, the algorithm effectively identifies and extracts individual columns from the image (Fig. 7).

Algorithm 1 Algorithm for extracting columns from the page

Input : Raw handwritten image, I_g.
Output : Constituent columns of the Page.
procedure : PRE-PROCESS THE IMAGE (BINARIZATION, DESKEW, MORPHOLOGICAL OPENING)
    $Clean\_Image(I_c) \leftarrow Preprocessing(I_g)$
    // Apply Vertical Projection
    $V_p \leftarrow VerticalProjection(I_c)$
    // Detect Separation Region Indices (SI) where the $V_p = 0$
    $SI \leftarrow V_p = 0$
    // Group all the SI
    Contiguous Regions $(C_r) \leftarrow Continuous(SI)$
    // Determine the Separation region (S_r)
    $S_r \leftarrow C_r > Threshold$
    // Determine the length of the Separation Region $(l(S_r))$
    for $i \leftarrow 1$ `to` $len(S_r)$ **do**
        // Find the minimum (Column _ min) and maximum (Column _ max) index of the page column
        $C\_min(i) \leftarrow min(S_r)(i)$
        $C\_max(i) \leftarrow max(S_r)(i)$
        // Adding the column to the column list.
        $List(Columns).Add(Column(i))$
    end for
    return $Listof(Columns)$
end procedure

After this, the horizontal projection is used to extract the lines within the text documents (Algorithm 2). The segmentation process divides an input image into lines of text based on the horizontal axis and a specified "cut" threshold. It identifies a sequence of consecutive non-zero values, which signify regions containing text. When such a sequence reaches the "cut" threshold, it is recognized as a line segment, and the corresponding text region is extracted. The default "cut" value of 3 is used to ensure the images are typically separated by more than three consecutive spaces or gaps. The corresponding lines segmented from the text documents of both the languages are shown in Figs. 8 and 9 respectively.

Algorithm 2 Algorithm for segmenting lines from the extracted columns

Input : Column Image/Segments, C_{img}.
Output: List of lines segments.
procedure :
    // Apply Horizontal Projection
    $H_p \leftarrow Horizontalprojection(C_{img})$
    // Detect Separation Region Indices (SI) where the $H_p = 0$
    $SI \leftarrow H_p = 0$
    // Group all the SI
    Contiguous Regions $(C_r) \leftarrow Continuous(SI)$
    // Extract the column segments and add them to the list.
    for $i \leftarrow 1$ `to` $len(S_r)$ **do**
        // Find the start and end index of the segment
        $Seg_{row_{start}}(i) \leftarrow min(S_r)(i)$
        $Seg_{row_{end}}(i) \leftarrow max(S_r)(i+1)$
        // Crop the segments using these indices
        $Seg(i) = Crop(C_{img}, Seg_{row_{start}}(i), Seg_{row_{end}}(i))$
        Add the cropped segments into the segment list (L_seg)
        $L\_seg.Add(Seg(i))$
    end for
    return $List\_of(Line\_segments)$
end procedure

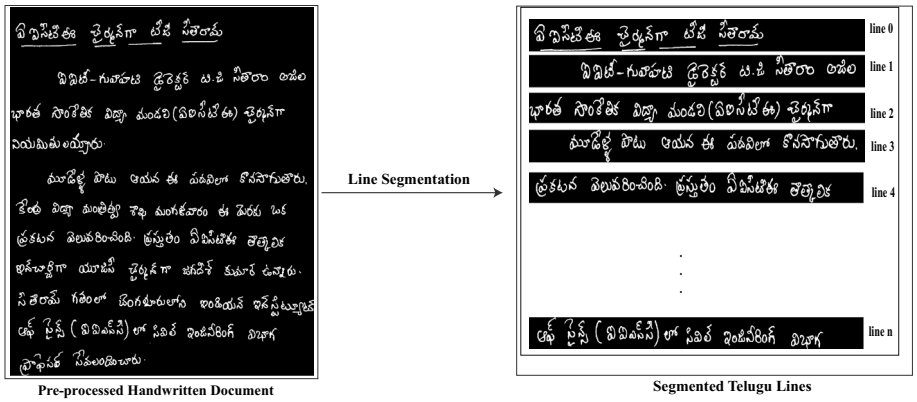

Fig. 9 Pre-processed Telugu document and the corresponding segmented lines obtained after line segmentation

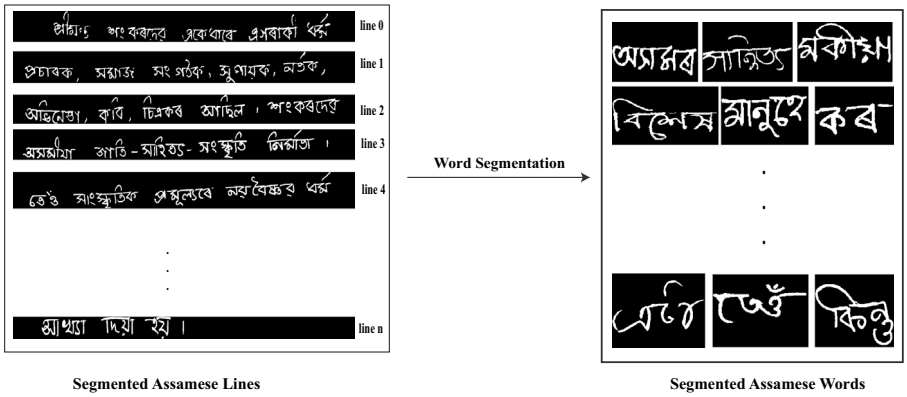

Fig. 10 Isolated Assamese lines segmented into words and characters

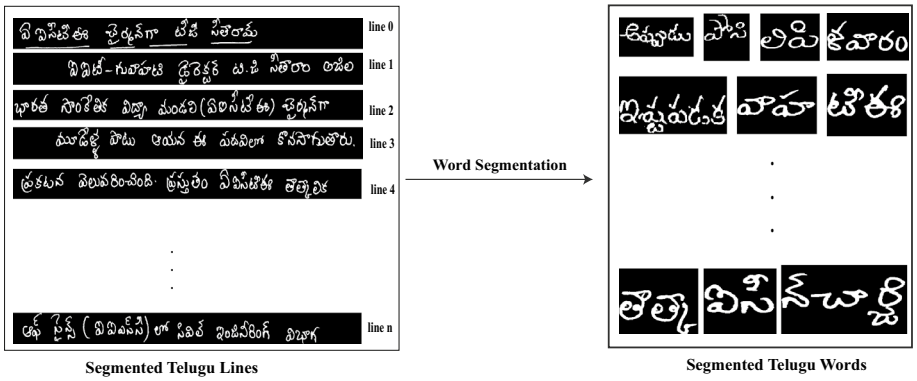

Fig. 11 Isolated Telugu lines segmented into words and characters

3.3.2 Word and character segmentation

Word segmentation, on the other hand, is performed within each of the obtained line segments. This approach takes the isolated lines as input and segments them into individual constituent words. Here, the line images are segmented along the vertical axis with another "cut" parameter (Qaroush et al. 2022). This parameter thus specifies the minimum number of consecutive empty vertical columns required to consider a gap between words. The word and character segmentation approach yields constituent words, characters, punctuation marks, and other formatting elements, markers, annotations, or punctuation marks present within the text. This is so because this approach analyses the gap between word boundaries which constitute these components. The lines and the corresponding words and characters extracted from each line are shown in Figs. 10 and 11.

3.3.3 Raster scanning

The words isolated from the previous word and character segmentation are now subjected to a Raster Scanning object detection approach to extract the constituent characters

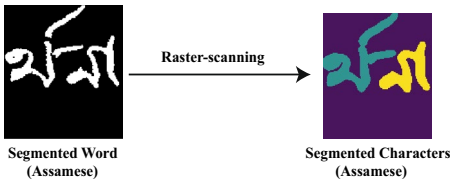

Fig. 12 Results of the raster scanning method applied to segment an Assamese word into its constituent characters

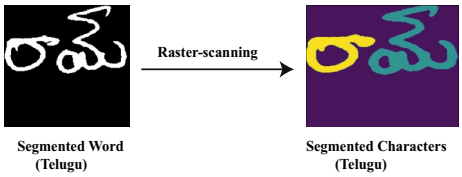

Fig. 13 Results of the raster scanning method applied to segment a Telugu word into its constituent characters

within them. The compound and composite characters will be considered isolated characters in this approach.

A Gaussian smoothing filter with a relatively small blur parameter, specifically a blur radius of 2, is employed. We opted for a smaller blur radius to balance the smoothing effect, maintaining crucial details and finer image features.

Connected component labeling groups adjacent foreground pixels (pixels with a value of 1) together to form a connected region. Each connected region is assigned a unique label, and this label is assigned to all pixels within that region. This label assignment is done so that pixels belonging to the same object have the same label, and pixels belonging to different objects have different labels. This approach yields—isolated characters, touching characters, and overlapping characters. Figures 12 and 13 shows a word obtained from the previous segmentation and the corresponding constituent characters obtained by applying the Raster Scanning object detection method.

Thus, the final work-flow diagram of the tow-down segmentation approach is shown in Fig 14. After applying pre-processing techniques to the handwritten documents, we employ a horizontal projection approach to isolate the lines of text. These isolated lines are the foundation for extracting individual words and characters through our word and character segmentation method. This process isolates characters, words, and additional elements like punctuation marks, markers, and diacritics. Subsequently, we direct the isolated words through the Raster scanning object detection approach, enabling us to identify and extract the constituent characters from within these isolated words.

4 Experimental results

In our effort to perform a comprehensive top-down segmentation on the gathered handwritten Assamese and Telugu text, we were able to segment the characters from the handwritten texts. Our segmentation process was conducted hierarchically, moving from the top-most level (documents) to the bottom-most level (characters).

4.1 Line segmentation results

Out of a total of 1145 and 1140 lines present for the Assamese and Telugu handwritten documents, our approach successfully extracted 1066 and 1073 lines accurately. The final accuracy of line segmentation is the average of segmentation accuracy of Assamese and Telugu texts into lines. Consequently, the average accuracy for line segmentation and recognition is 93.61%. These are shown in Table 1.

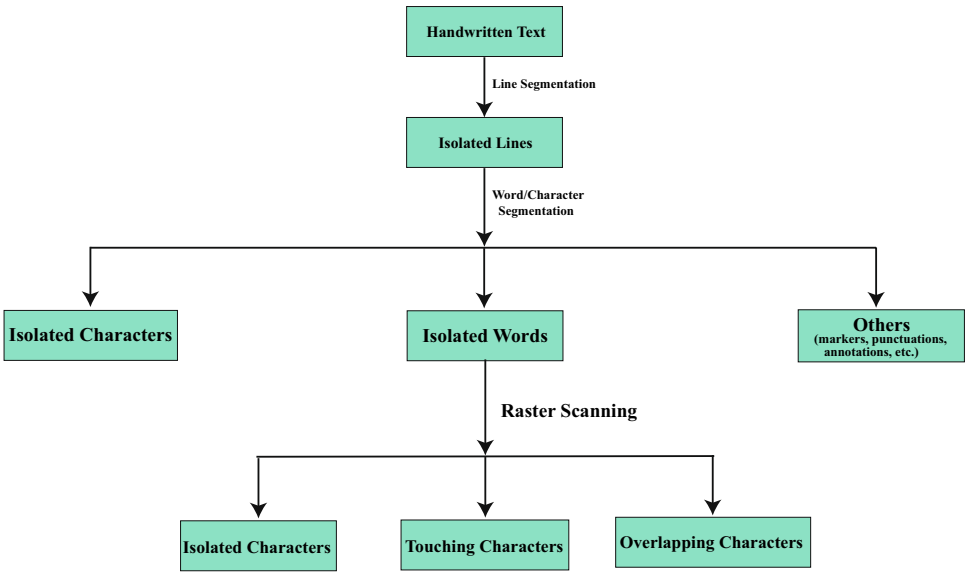

Fig. 14 Workflow diagram of the top-down segmentation approach: This diagram illustrates the hierarchical character segmentation levels devised for extracting characters from handwritten documents

Table 1 Line segmentation results of Assamese and Telugu handwritten text datasets

Language	Actual no. of lines	Correctly segmented lines	Accuracy
Assamese	1145	1066	93.10%
Telugu	1140	1073	94.12%
Total	2285	2139	**93.61%**

Table 2 Word segmentation results of our Assamese and Telugu handwritten text datasets

Language	Actual no. of words	Isolated words	Isolated characters	Others	Accuracy (word segmentation)
Assamese	9300	7945	947	408	85.43%
Telugu	10,766	9312	1002	452	86.49%
Total	20,066	17,257	1949	860	**85.96%**

4.2 Word segmentation results

With the lines obtained from the previous step and applying the second level of segmentation, we were able to obtain isolated characters, words, and other components (Fig 14). From a total of 9, 300 and 10, 766 words in the Assamese and Telugu lines, 7, 945 and 9, 312 words were successfully extracted by the approach. The final accuracy of word segmentation is the average of segmentation accuracy of Assamese and Telugu lines into words. Hence, the average word segmentation accuracy is 85.96%. The results are shown in Table 2.

The isolated characters will be considered in the final character segmentation results.

4.3 Character segmentation results

Our approach employs connected component labeling and distinct color representations to identify individual characters within a word image, including consonants and compound characters. The final, accurately segmented characters combine the isolated characters extracted through the vertical projection method (Sect. 3.3.2) and those identified through the object detection technique during raster scanning (Sect. 3.3.3). The final accuracy of character segmentation is the average of the character segmentation accuracy of Assamese and Telugu words into characters. Hence we get the average character segmentation accuracy of 88.74%. The results are shown in Table 3.

The total segmented character count is a sum of the isolated characters obtained through the word segmentation method and the raster scanning approach. This results in a combined count of 20,226 characters for the Assamese language, comprising 1959 characters from word segmentation and an additional 18,267 characters from the raster scanning approach. Similarly, for the Telugu language, the total character count is 23,377, comprised of 2467 characters from word segmentation and 20,910 characters from raster scanning. Hence, when considering the total segmented character count, we find that 20,226 characters were extracted from the original 22,908 characters for the Assamese language. Similarly, in the case of the Telugu, 23,377 characters were obtained from the initial 26,223 characters. This translates to an average character segmentation accuracy of 88.74%.

5 Discussion

We utilized projection profile-based and connected component-based techniques to achieve efficient character segmentation in handwritten document images of Assamese and Telugu. Initially, the handwritten documents were isolated into constituent lines by the horizontal projection approach. This proved versatile and capable of handling diverse forms of handwritten text, including cases with slight non-parallel lines and variations in the presence and absence of the "Matra" component. It's worth noting that, despite these remarkable results, we encountered some mis-segmentations, specifically pointing towards *under-segmentation*.

Under-segmentation occurs when the segmentation method fails to divide the text lines into individual lines or words correctly. In line segmentation, multiple lines of text are erroneously grouped as a single unit. This can result in a loss of readability and pose challenges for subsequent text analysis tasks, i.e., word and character segmentation. Under-segmentation occurred in our study because of the following scenarios: The first one is that some text lines are closely spaced, making it difficult for the segmentation algorithm to distinguish between them (Fig. 15). Secondly, while writing, the upper line of the current line touches the lower zone of the previous line. Thus, two or more successive lines were

Table 3 Character segmentation results from the Assamese and Telugu words

Language	Total characters	Characters (word segmentation)	Characters (raster scanning)	Total segmented characters	Character segmentation accuracy
Assamese	22,908	1959	18,267	20,226	88.29%
Telugu	26,223	2467	20,910	23,377	89.14%
Total	49,131	4426	39,177	43,603	**88.74%**

Table 4 Performance comparison of our approach with current literature

Literature	Language	Approach	Accuracy
Bag et al. (2011)	Bangla	Vertical characterization	96.04%
Tamhankar and Masalkar (2020)	Marathi	Dual thresholding	67%
Jindal and Ghosh (2023)	Devanagari and Maithili	Horizontal zoning	98.35% and 98.65%
Proposed	Assamese and Telugu	Vertical projection + raster scanning	**88.29%** and **89.14%**

Bold values indicate that our proposed top-down character segmentation approach performs better than the latest character segmentation approaches available in literature

Fig. 15 Closely spaced Assamese (left) and Telugu (right) text lines identified as a single line

Fig. 16 Slanted Assamese (left) and Telugu (right) text lines identified as a single line

Fig. 17 Irregularly aligned Assamese (left) and Telugu (right) text lines identified as a single line

segmented as isolated lines (Fig. 16). Lastly, certain text lines may exhibit irregularities even after applying deskewing. These irregularities can lead to situations where a text line slants upward as it progresses from left to right. Consequently, the end of one text line may overlap with or touch the text from the previous line (Fig. 17).

As a result of these scenarios, the projection profile technique proposed in our work failed to identify these segments as distinct lines correctly. Instead, it may interpret them as single entities or lines. In essence, the inherent variability in the handwriting style made it challenging.

It's important to note that the under-segmentation issues observed at the line segmentation level impact word segmentation. Occasionally, two adjacent lines may be under-segmented into a single unit, resulting in one word being comprised of what should be two separate words from two separate lines. This underscores the importance of efficient line segmentation, as it forms the foundation for accurate word segmentation. Addressing under-segmentation at the line level will likely lead to improvements in word segmentation, ultimately enhancing the overall accuracy of our document analysis process.

One of the notable achievements of this method is its accuracy in segmenting lines of texts, words, and, finally, characters encompassing various font styles and sizes in both languages. This approach works best in addressing the specific challenge we set out to overcome: the presence and absence of the mantra component. To comprehensively assess its performance, we carefully selected two datasets in Assamese and Telugu-one containing the mantra component and the other without the mantra- and effectively segmented text in both datasets.

A comparison of our work with existing literature on character segmentation of handwritten documents is shown in Table 4. Our approach may not claim the highest accuracy, but it addresses the complex challenges associated with handwritten character segmentation of the two languages under consideration. The achieved accuracies in the proposed approach align with the outcomes reported in existing literature for character segmentation across different languages. While it may not have the highest accuracy in this comparison, the approach demonstrates competitive performance in addressing the complexities of handwritten character segmentation.

This novel approach can be considered competitive within the context of character segmentation in handwritten documents. Nevertheless, it's worth acknowledging that the field of character segmentation remains open to further enhancements and refinements, given its intricate nature.

6 Conclusion and future work

We have introduced a novel character segmentation approach for handwritten documents, specifically focusing on the regional languages of Assam and Andhra Pradesh. Our top-down segmentation approach, combining horizontal and vertical projection techniques, has proven effective in accurately segmenting lines, words, and characters, even in the presence or absence of the Matra component. Another significant contribution of our approach is the utilization of the Raster Scanning object detection approach, which further tries to obtain the constituent characters from the isolated words. Thus the character segmentation results are obtained from both the vertical segmentation as well as the Raster scanning object detection approach, which enhanced the character segmentation accuracy in our study. We achieved character recognition accuracy rates of 88.29% and 89.14% in both Assamese and Telugu datasets, showcasing the versatility of our method.

However, by neglecting under-segmented text lines, there was a reduction in the count of extracted characters and words. Additionally, extracting constituent characters from touching and overlapping characters could potentially increase the overall count of extracted characters.

In the future, under-segmentation scenarios can be handled by developing better experimentation ideas. The literature provides a wealth of overlapping and touching character segmentation methods that can be integrated into our existing approach to enhance its performance.

We have observed that vowels, consonants, and even compound characters (combination of vowels and consonants) are obtained at the bottom-most segmentation level. Relevant datasets of compound characters of the regional languages also need to be collected. This is needed since after segmenting the words into characters, compound characters are also identified as characters. Hence, for classifying and recognizing them further, such datasets will be relevant.

Data availability The Assamese and Telugu handwritten text dataset is publicly available at the IEEE dataport. Relevant link for the dataset is provided in Sect. 3.1.

Declarations

Conflict of interest The authors declare no conflict of interest in any part of the work presented in the manuscript.

Ethical approval In accordance with ethical standards, our research obtained informed consent from participants and followed all necessary ethical review procedures.

References

Abdulhussain SH, Mahmmod BM, Naser MA et al (2021) A robust handwritten numeral recognition using hybrid orthogonal polynomials and moments. Sensors 21(6):1999

Ahamed P, Kundu S, Khan T et al (2020a) Handwritten Arabic numerals recognition using convolutional neural network. J Ambient Intell Humaniz Comput 11:5445–5457

Ahmad R, Naz S, Afzal MZ et al (2020b) A deep learning based Arabic script recognition system: benchmark on Khat. Int Arab J Inf Technol 17(3):299–305

Ali AAA, Suresha M (2020) Survey on segmentation and recognition of handwritten Arabic script. SN Comput Sci 1(4):192

Bag S, Bhowmick P, Harit G et al (2011) Character segmentation of handwritten Bangla text by vertex characterization of isothetic covers. In: 2011 Third national conference on computer vision, pattern recognition, image processing and graphics, IEEE, pp 21–24

Bangare SL, Dubal A, Bangare PS et al (2015) Reviewing Otsu's method for image thresholding. Int J Appl Eng Res 10(9):21777–21783

Barakat BK, Droby A, Alaasam R et al (2021) Unsupervised deep learning for text line segmentation. In: 2020 25th International conference on pattern recognition (ICPR). IEEE, pp 2304–2311

Batchas BM, Shahid M (2021) The need of a digital typeface for Assamese script. In: International conference of the Indian society of ergonomics. Springer, pp 1599–1610

Bose M (1989) Social history of Assam: being a study of the origins of ethnic identity and social tension during the British period, 1905–1947. Concept Publishing Company, India

Chatterjee I, Ghosh M, Singh PK et al (2019) A clustering-based feature selection framework for handwritten indic script classification. Expert Syst 36(6):e12459

Cheikhrouhou A, Kessentini Y, Kanoun S (2021) Multi-task learning for simultaneous script identification and keyword spotting in document images. Pattern Recogn 113:107832

Chen K, Seuret M, Hennebert J et al (2017) Convolutional neural networks for page segmentation of historical document images. In: 2017 14th IAPR international conference on document analysis and recognition (ICDAR). IEEE, pp 965–970

Chen X, Jin L, Zhu Y et al (2021) Text recognition in the wild: a survey. ACM Comput Surv (CSUR) 54(2):1–35

Chirimilla R, Vardhan V (2022) A survey of optical character recognition techniques on indic script. ECS Trans 107(1):6507

Dutta P, Muppalaneni NB (2022) A survey on image segmentation for handwriting recognition. In: Third international conference on image processing and capsule networks: ICIPCN 2022. Springer, pp 491–506

Dutta P, Muppalaneni NB (2024) Assamese and Telugu handwritten text dataset. 10.21227/3ycm-px23

Dutta A, Garai A, Biswas S et al (2021) Segmentation of text lines using multi-scale cnn from warped printed and handwritten document images. International Journal on Document Analysis and Recognition (IJDAR) 24(4):299–313

Girdher H, Sharma H, Gupta A (2022) Comprehensive survey on Devanagari OCR. Available at SSRN 4033489

Grüning T, Leifert G, Strauß T et al (2019) A two-stage method for text line detection in historical documents. Int J Docum Anal Recogn (IJDAR) 22(3):285–302

Inunganbi S, Choudhary P, Manglem K (2021) Meitei Mayek handwritten dataset: compilation, segmentation, and character recognition. Vis Comput 37(2):291–305

Jindal A, Ghosh R (2023) Word and character segmentation in ancient handwritten documents in Devanagari and Maithili scripts using horizontal zoning. Expert Syst Appl 225:120127

Joseph S (2022) Advanced digital image processing technique based optical character recognition of scanned document. J Innov Image Process 4(3):195–205

Kaur RP, Kumar M, Jindal M (2022) Performance evaluation of different features and classifiers for Gurumukhi newspaper text

recognition. J Ambient Intell Human Comput. https://doi.org/10.1007/s12652-021-03687-8
Krishna MV, Ram KJ (2021) Digitization, preservation and character recognition in ancient documents using image processing techniques—a review. Int J Commun Comput Technol 9(1):23–26
Kundu S, Paul S, Bera SK et al (2020) Text-line extraction from handwritten document images using gan. Expert Syst Appl 140:112916
Lee AW, Chung J, Lee M (2021) Gnhk: A dataset for English handwriting in the wild. In: Document analysis and recognition–ICDAR 2021: 16th International conference, Lausanne, Switzerland, September 5–10, 2021, Proceedings, Part IV vol 16. Springer, pp 399–412
Li D, Wu Y, Zhou Y (2021) Linecounter: learning handwritten text line segmentation by counting. In: 2021 IEEE international conference on image processing (ICIP). IEEE, pp 929–933
Malik SA, Maqsood M, Aadil F, et al (2020) An efficient segmentation technique for urdu optical character recognizer (ocr). In: Advances in information and communication: proceedings of the 2019 future of information and communication conference (FICC), vol 2. Springer, pp 131–141
Mioulet L, Garain U, Chatelain C et al (2015) Language identification from handwritten documents. In: 2015 13th International conference on document analysis and recognition (ICDAR). IEEE, pp 676–680
Obaidullah SM, Santosh K, Halder C et al (2019) Automatic indic script identification from handwritten documents: page, block, line and word-level approach. Int J Mach Learn Cybern 10:87–106
Pastor-Pellicer J, Afzal MZ, Liwicki M, et al (2016) Complete system for text line extraction using convolutional neural networks and watershed transform. In: 2016 12th IAPR workshop on document analysis systems (DAS). IEEE, pp 30–35
Qaroush A, Jaber B, Mohammad K et al (2022) An efficient, font independent word and character segmentation algorithm for printed Arabic text. J King Saud Univ Comput Inf Sci 34(1):1330–1344
Rahman AA, Hasan MB, Ahmed S et al (2022) Two decades of Bengali handwritten digit recognition: a survey. IEEE Access 10:92597–92632
Rajyagor B, Rakholia R (2021) Tri-level handwritten text segmentation techniques for Gujarati language. Indian J Sci Technol 14(7):618–627
Renton G, Chatelain C, Adam S et al (2017) Handwritten text line segmentation using fully convolutional network. In: 2017 14th IAPR International conference on document analysis and recognition (ICDAR). IEEE, pp 5–9
Singh G, Sachan MK (2020) An unconstrained and effective approach of script identification for online bilingual handwritten text. Natl Acad Sci Lett 43(5):453–456
Singh A, Bacchuwar K, Bhasin A (2012) A survey of ocr applications. Int J Mach Learn Comput 2(3):314
Singh S, Garg NK, Kumar M (2023) Feature extraction and classification techniques for handwritten Devanagari text recognition: a survey. Multimed Tools Appl 82(1):747–775
Srivastava S, Verma A, Sharma S (2022) Optical character recognition techniques: a review. 2022 IEEE international students' conference on electrical, electronics and computer science (SCEECS). IEEE, pp 1–6
Suleyman E, Hamdulla A, Tuerxun P et al (2021) An adaptive threshold algorithm for offline uyghur handwritten text line segmentation. Wireless Netw 27:3483–3495
Tamhankar PA, Masalkar KD et al (2020) A novel approach for character segmentation of offline handwritten Marathi documents written in Modi script. Proc Comput Sci 171:179–187
Ukil S, Ghosh S, Obaidullah SM et al (2020) Improved word-level handwritten indic script identification by integrating small convolutional neural networks. Neural Comput Appl 32(7):2829–2844
Yousef M, Bishop TE (2020) Origaminet: weakly-supervised, segmentation-free, one-step, full page text recognition by learning to unfold. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 14710–14719
Zhou J, Wang F, Xu J et al (2019) A novel character segmentation method for serial number on banknotes with complex background. J Ambient Intell Human Comput 10:2955–2969
Zouari R, Boubaker H, Kherallah M (2019) Multi-language online handwriting recognition based on beta-elliptic model and hybrid TDNN-SVM classifier. Multimed Tools Appl 78(9):12103–12123