Journal of
Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.7752024

AUTOSEG: AN AUTOMATED SEGMENTATION APPROACH FOR TYPEWRITTEN GURMUKHI CHARACTER RECOGNITION

Gurvir Kaur[1], Dr. Ajit Kumar[2]
[1]Department of Computer Science, Punjabi University, Patiala, India;
[2]Multani Mal Modi College, Patiala, India

Abstract: Background: The increasing demand for digital media influenced many fields. Most of the content is now available digitally for easy access to every user. India is a multilingual country and rich in culture. Most of the historical Vedas, Granths, literature books, poetries, etc., are in the local languages, so now there is a need to develop the systems that help to secure this history or culture in a digital form so people from other cultures can also understand or read it by translating it in their language. Many researchers have worked on different languages, but recognition systems still need improvement. The Gurmukhi language is one of them. Though systems were developed for the Gurmukhi language, the samples were either handwritten or printed. Much of the literary data is present in the typewritten form, as typewriters were very popular earlier, so there is a need to develop a system that automatically deals with this data.

Objective: In this work, an automated segmentation approach (AutoSeg) is proposed that takes an input of typewritten Gurmukhi text in an image form and segments it.

Methods: The proposed approach (AutoSeg) is divided into two phases: pre-processing to enhance the quality of the sample image and then the segmentation phase to segment all the characters from the sample image. This automated system will help derive a system that can recognize the character accurately to use in the future. The proposed AutoSeg uses different image processing methods to segment the lines, words, and characters from the given image samples. The proposed approach also deals with the issue of broken and touching characters.

Results: The proposed approach is tested using our collected database of books, thesis, poetry documents, etc., and segmentation accuracy is measured at each level. It has been found that the proposed approach achieved an accuracy of more than 90% for each line, word, and character segmentation, proving the approach's effectiveness.

Conclusion: The novel automated segmentation approach is proposed that effectively segments the Gurmukhi typewritten characters and deals with the problems like broken or touching characters. The results also show the effectiveness of this segmentation approach. Hence, it can be utilized soon for Gurmukhi typewritten character recognition systems.

Keywords: Automatic Segmentation, Line, Word, Character, Gurmukhi Text, Typewritten.

1. INTRODUCTION

Digitalization is becoming increasingly prevalent in people's lives and has a variety of implications as technology advances. One aspect of digitization that has a big impact is text recognition systems [1]. The effectiveness of various text extraction techniques developed by researchers was tested across a range of languages [2]. Even in remote places in India, a multilingual country, languages are changing [3]. The researchers' study was less extensive in many languages than it was in many others. Additionally, some researchers provided recognition algorithms for Indian languages [4]. One of these is the Gurmukhi language, and this research's focus was on the Gurmukhi text recognition system [5].

The typewritten text recognition system examines samples that have been taken using photographs or scanned with scanners [6]. A typeface that is significantly similar in size and shape is used in examples of typewritten texts [7]. Even still, assessing the quality of outdated typewritten documents is more difficult than it is for printed or other types of documents [8]. The typewritten text's quality is also subpar and inconsistent [9]. Many documents have also deteriorated because of the utilized ink and paper. Additionally, the typewritten text frequently contains irregular letters, some of which are either darker or fainter than others depending on how firmly the typewriter key was pressed [10]. Working with typewritten publications is challenging compared to other machine-produced publications [11]. Typewritten document recognition systems also use the standard recognition system processes of (a) pre-processing, (b) segmentation, (c) feature extraction, (d) classification, and (e) post-processing (if necessary) [12].

Documents are divided into lines, characters, or words in order to boost recognition rates [13]. It frequently applies to texts and distinguishes between text and visuals, illustrations, or figures. It works with two categories: segmentation on the outside and inside [14]. Using external/holistic segmentation, it is often possible to separate any complicated text image or language form into text and non-text components[15]. For instance, external segmentation is a key component of page layout analysis, also known as document analysis, which is further divided into the structural and functional analysis. Internal segmentation, on the other hand, divides the words in the image into distinct characters or symbols as a set of pixels with some linked features that are more crucial [16].

2. RELATED WORK

The segmentation procedure is crucial for improving the precision of the recognition systems. The researchers must therefore pay special attention to this important step in order to offer a successful strategy. Different segmentation strategies for text recognition systems for languages other than English were presented by various scholars. This section talks about the techniques that are currently in use. In-text recognition systems collect samples from a variety of papers, books, documents, newspapers, and other media. When samples contain more than just text data, object removal becomes the researchers' primary priority. A hybrid solution that divides the picture region using run-length smoothing techniques and projection profile approaches has been proposed by Kaur et al. [17]. Goyal et al. [18] proposed a novel segmentation technique to separate lines, words, and characters while keeping this in mind. Pre-processing, Line Segmentation, Word Segmentation, and Character Segmentation are a few examples of data segmentation techniques that they used over their four main stages of work. They also thought about the problem of the damaged and touching characters of the typewritten Gurmukhi text recognition system. Ayesh et al. [8] recommended segmenting the lines using a projection-based approach. Another text line recognition method for line segmentation in historical materials was put out by Slimane et al. [19]. They used the Fisher method to binarize, and after that, they carried out morphological operations. Binarization, according to some academics, can be used to separate data and is thought to play a significant role in segmentation. For instance, Paulus et al's[20] line segmentation procedure used the best of four alternative binarization algorithms after initially employing them all. A further method for identifying and segmenting lines that are projection-based was suggested by Narang et al. [21]. The piecewise projection profile method was used to project the document images after dividing them into stripes of a set size. The same strategy, which uses unsupervised learning, was put forth by Barakat et al. [22] and other machine learning-based approaches, which are also well-liked.

Few academics think that word segmentation can greatly enhance the recognition system, especially for Gurmukhi text. Dahake et al.[23] proposed a bounding box and threshold-based technique that leverages the boxes' min and max position values to accomplish word segmentation. A new approach for segmenting words based on projection properties was proposed by Mahto et al. [24]. Text recognition systems have a difficult time segmenting characters, especially for Indian languages. Sahare and Dhok [25] assessed a character segmentation technique using Hindi and English. Tamhankara et al. [26] proposed the alternate character segmentation approach using handwritten Marathi language texts. The proposed approach was built on the projection profile approach, which successfully distinguished the characters using the VPP approach. Another approach for character segmentation based on black pixel values was proposed by Thakral and Manoj [27]. For the Devnagri language characters, this approach was made available. Li et al's [28] unique text segmentation approach is recommended for separating text from images with a complicated background or low resolution. Elharrouss et al.'s[29] approach for text segmentation as well as line and word analysis was recommended for historical Arabic manuscripts.

These systems' accuracy as they stand now is quite astounding. However, as most methods only utilize small datasets to generate highly accurate results, the performance of the presented methods must be assessed using a big dataset.

PROPOSED AUTOSEG APPROACH

The input image is divided into words, lines, and characters using the recommended AutoSeg approach. For line segmentation, a technique based on the horizontal projection profile (HPP) is employed [31]. Utilizing the average peak value from the HPP findings, the peak's position will first be established. Additionally, these locations split the areas based on their average values. The height of the separated portions is next measured, and if it is higher than the average height, additional peaks are created. The average value of each zone is then measured. The words for each line are then extracted using the connected component technique [15]. For each Segmented Line, dilation [32] is first performed. In this case, dilatation fills the space between the characters as a result of the broken characters, ensuring proper segmentation. The area of the words is then divided in accordance with their associated components [33]. The thinning and connected component approach is then used to separate the characters for each word. This method first performs morphological thinning [34] for each segmented word. In this instance, the characters are touching, hence thinning is done to separate the characters for accurate segmentation. Characters are split using related components after the upper region linking the word depending on the maximum number of white pixels is removed. The proposed AutoSeg approach's pseudo-code is provided below:

Algorithm 1: **AutoSeg approach**

Input: **Typewritten text image**

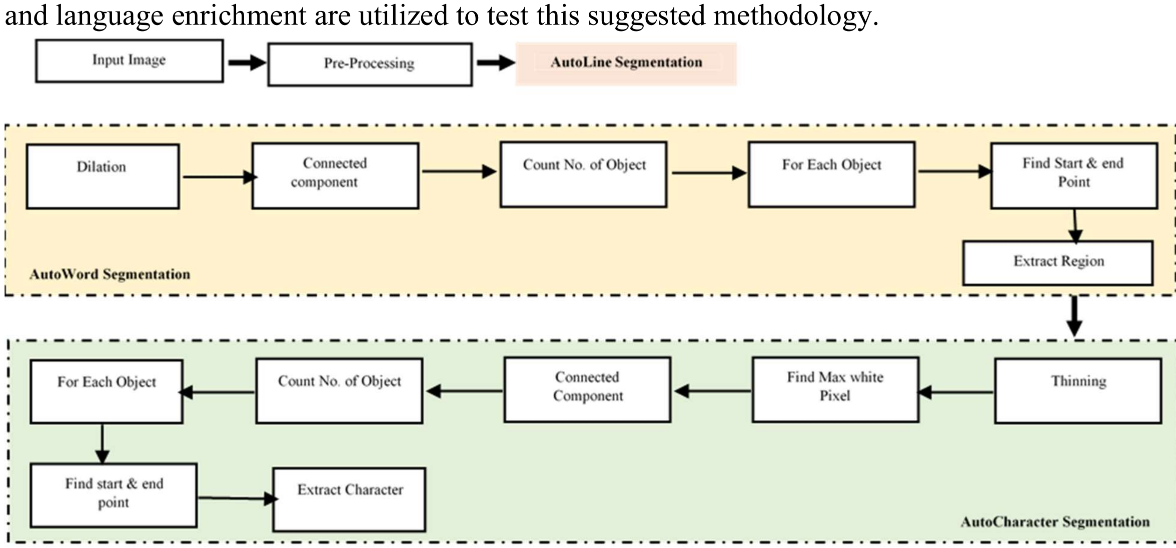
Output: **Segmented Character**	
(1)	*Start*
(2)	*input_image [ ] ← read(typewritten_text_image)*
(3)	*input_image(input_image==0) = 1*
(4)	*input_image(input_image==255) = 0*
(5)	*HPP[ ] ← sum(input_image,1)*
(6)	*Avg_HPP ← average(HPP[ ])*
(7)	*for each val in HPP*
(8)	*if val < Avg_HPP*
(9)	*region_1 ←region separation(input_image[val])*
(10)	*else*
(11)	*go to step (5)*
(12)	*end of if*
(13)	*end of for*
(14)	*h ← average_height(region_1)*
(15)	*for each val in h*
(16)	*if val <h*
(17)	*line_region ← region separation(input_image[val])*
(18)	*else*
(19)	*go to step (7)*
(20)	*end of if*
(21)	*end of for*
(22)	*for each region in line_region*
(23)	*se ← strel('disk',3)*
(24)	*dilate_out ← imdilate(region, se)*
(25)	*cc ← connected_components(dilate_out)*
(26)	*n ← size(cc)*
(27)	*for each obj in n*
(28)	*start_pt ← min(obj)*
(29)	*end_pt ← max(obj)*
(30)	*word_region ← obj(start_pt:end_pt)*
(31)	*end of for*
(32)	*for each region in word_region*
(33)	*thin_out ← bwmorph(region, 'thin')*
(34)	*out ← find a row with maximum white pixels and remove it (thin_out)*
(35)	*cc1 ← connected_components(out)*
(36)	*n ← size(cc1)*
(37)	*for each obj in n*
(38)	*start_pt ← min(obj)*
(39)	*end_pt ← max(obj)*
(40)	*char_region ← obj(start_pt:end_pt)*
(41)	*end of for*
(42)	*end of for*
(43)	*end of for*
(44)	*End*

Each axis' projection profile is calculated separately. The horizontal projection profile is the projection profile of an image along the horizontal axis. The sum of all the column pixel values inside a row is used to determine it. As was already indicated, the method additionally makes use of morphological operations, a collection of non-linear procedures for the shape component of an image. These approaches involve positioning structural elements in all possible locations and comparing them to surrounding pixels. The most common processes are dilation and erosion, which are used for growing and thinning, respectively. This study employs dilatation to deal with the problem of broken characters and accurate segmentation. Dilation enlarges the boundaries of objects in an image by using pixels. In order to solve the problem of touching characters and accurate character segmentation, thinning is also used. Using morphological approaches, binary images can be thinned to get rid of particular foreground pixels. Although it can be used in a variety of other circumstances, skeletonization is one in which it excels. is the connected component approach, which links pixels based on intensity values to produce a set of connected components. The image's constituent parts are groups of identical pixels connected to one another using either 4- or 8-pixel connectivity. While 4-pixel connectivity groups all pixels that come into touch with one another on any of its four faces, 8-pixel connections join pixels along any face or corner. The approach employed for this work is based on eight connected neighbors.

PROPOSED WORK

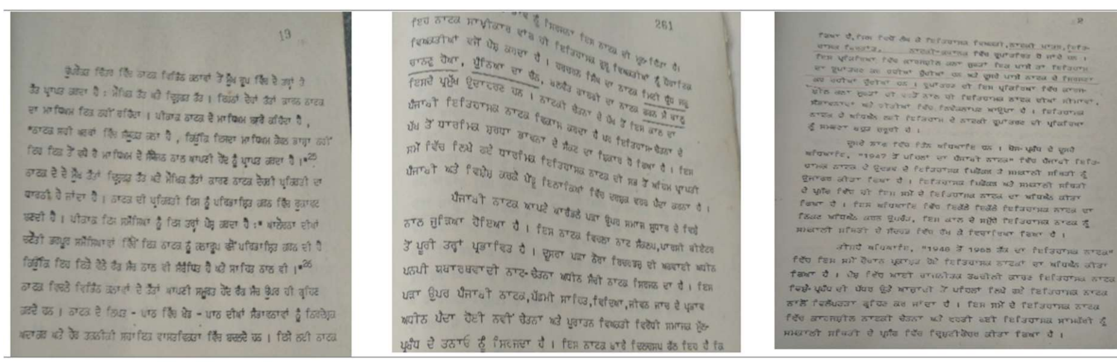
Separating the typewritten Gurmukhi characters for recognition systems is the prime objective of the proposed work. An AutoSeg technique, as mentioned in the section before, is suggested for this. In this study, the AutoSeg method was used to propose a system for segmenting characters from image data samples. As a result of the data being gathered using a camera and the samples being camera-captured images, the images have various quality flaws, such as a blurring effect, shadow, and contrast difficulties. To solve this, the suggested work is split into two phases: (a) Image Quality Enhancement (pre-processing) and (b) Segmentation Phase. The first phase of pre-processing image samples employs noise reduction, contrast enhancement, and smoothing; the second phase employs image binarization, skew correction, and three-level segmentation, which comprises line [15], word, and character level segmentation. The suggested system's design is shown in Figure 1, and its specifics are as follows:

Step 1: Data Acquisition: A large collection of typewritten Gurmukhi documents from various sources has been assembled. The dataset includes images of Gurmukhi typewritten text-based documents collected from universities, colleges, and other sources using a smartphone camera. This dataset comprises full images of typewritten religious texts, stories, poems, and novels in Gurmukhi. 1000 samples from the more than 5,000 document images used for OCR research and language enrichment are utilized to test this suggested methodology.



Step 2: Pre-processing: As seen in figure 2, the collected image samples exhibit several problems with blurring, shadows, and contrast. Pre-processing approaches, where various filters, processes, or strategies can be used depending on the needs of the samples, can overcome all these problems. So, noise removal, contrast enhancement, and smoothing are pre-processing techniques [15] in this work to prepare image samples. It aids in acquiring high-quality samples and enhances system performance. Since the samples were taken with a camera device, they are unavoidably present. Therefore, the first primary goal is to eliminate sample noise brought on by extraneous conclusions. Weiner [35] and anisotropic [36] filters are utilized for this purpose, and a hybrid filter is then developed employing these two filters. Finally, an improved image is produced by sequentially combining the wiener and anisotropic filters, which helps preserve critical details like edges even after removing noise. Because some of the gathered samples have poor contrast, the contrast of the sample is adjusted using the contrast enhancement approach to create a qualitative image. In order to equalize contrast, 1% of the data in this work is saturated at both high and low intensities. This technique is known as intensity saturation. The sharpening effect is seen in the samples after the noise has been removed and the contrast has been adjusted. So, to manage the excess of sharpness, samples must be smoothed; in this work, median [37] and average [38] are employed. Combining the filters mentioned above to create a hybrid filter is another method of further improving this stage. It sequentially combines median and average filters to boost the system's performance. An example of the pre-processing phase with hybrid filters for noise removal and image smoothing is shown in figure 3.

Figure 1: Proposed System Architecture



(a) **Blurring** (b) **Shadow** (c) **Contrast**

Figure 2: Quality issues in the collected dataset

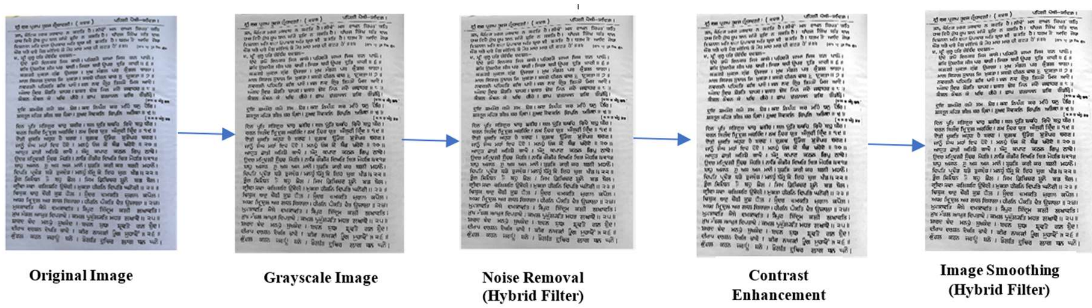


Figure 3: Pre-processing

The conversion of a colour or grayscale image into a binary image can be accomplished utilizing a variety of image processing techniques. This study uses the OTSU technique [39], which was chosen based on experiments [15], for binary conversion. An image will automatically transform into a binary image using this method. For a threshold that assists in differentiating the foreground and background regions, the single intensity value is returned. Utilizing the maximization of inter-class intensity variance and the minimizing of intra-class intensity variation, the absolute threshold value is determined. Earlier research projects [4], [9] that dealt with text recognition system segmentation used the OTSU technique. Skew correction is essential for text recognition as well since typewritten text typically has a skewness problem that impairs the accuracy of segmentation. After skew correction, the foreground text pixels are recovered, and the entropy-based projection profile approach [40] is used to compute the essential information for creating the final output images with the right angles [15].

Step 3: Segmentation: The automatic segmentation approach put forth in this study extracts the lines from the sample image that is provided, the words from each line, and finally the characters from the segmented words. The AutoSeg approach, which goes by that name, was covered in the previous section of the paper. The results of line, word, and character segmentation are shown in the following figure.

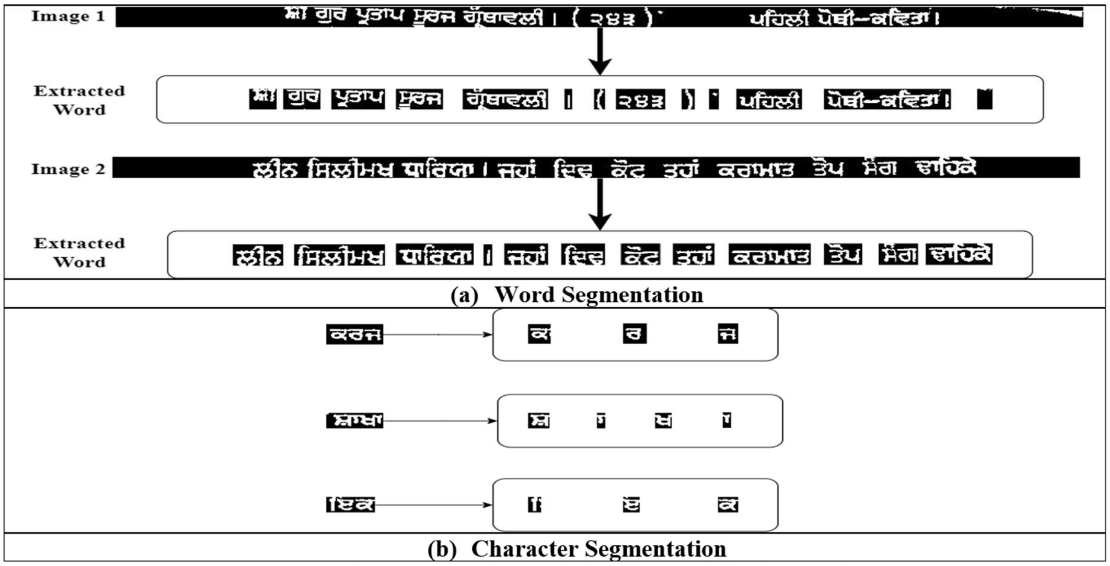


Figure 4: Segmentation Outcomes (a) Words from lines, and (b) Character from words

Figure 4: Segmentation Outcomes (a) Words from lines, and (b) Character from words

EXPERIMENTATION AND RESULT ANALYSIS

The effectiveness of the proposed approach is evaluated using a dataset of typewritten Gurmukhi thesis, narrative, poetry, fiction, and religious books. In total, 1,000 images were used in this study, divided into two datasets with 500 samples each. This project's first step is to increase the quality of the image sample because some of the acquired images have contrast, blurriness, and other issues. The quality of the images is therefore assessed after pre-processing, and the enhanced samples are then forwarded to the second phase, where segmentation, skew correction, and binarization are carried out. The outcomes of two phases—the Image Quality Enhancement phase and the Segmentation Phase—are also validated through experiments. When utilized to measure the quality of a picture, Peak Signal Noise Ratio (PSNR), Structural Content (SC), Laplacian Mean Square Error (LMSE), and Normalized Absolute Error (NAE) are commonly used parameters [41] that provide an accurate evaluation of the quality of the image sample. For instance, the PSNR assesses how well a picture can be represented by comparing its maximum power to the power of corrupting noise. Since lower noise is linked to higher PSNR, it should therefore be higher. Additionally, for photos of greater quality, SC, the sum of the squares of the original and recovered image pixel values, should have a better value. On the other hand, the LMSE and NAE error metrics should be as minimal as possible. Three pre-processing procedures, as mentioned in the preceding part, are employed in this study, and different filters are used for each technique. In the presented results (figure 5), five distinct methods—M1, M2, M3, M4, & M5—are utilized. These approaches were created by combining various pre-processing techniques as listed in the table below:

Table 1: Image quality Enhancement Methods

Table 1: Image quality Enhancement Methods

Method	Noise Removal	Contrast Enhancement	Image Smoothing
M1	Weiner Filter	Intensity Saturation Method	Median Filter
M2	Weiner Filter	Intensity Saturation Method	Average Filter
M3	Anisotropic Filter	Intensity Saturation Method	Median Filter
M4	Anisotropic Filter	Intensity Saturation Method	Average Filter
M5	Hybrid Filter	Intensity Saturation Method	Hybrid Filter

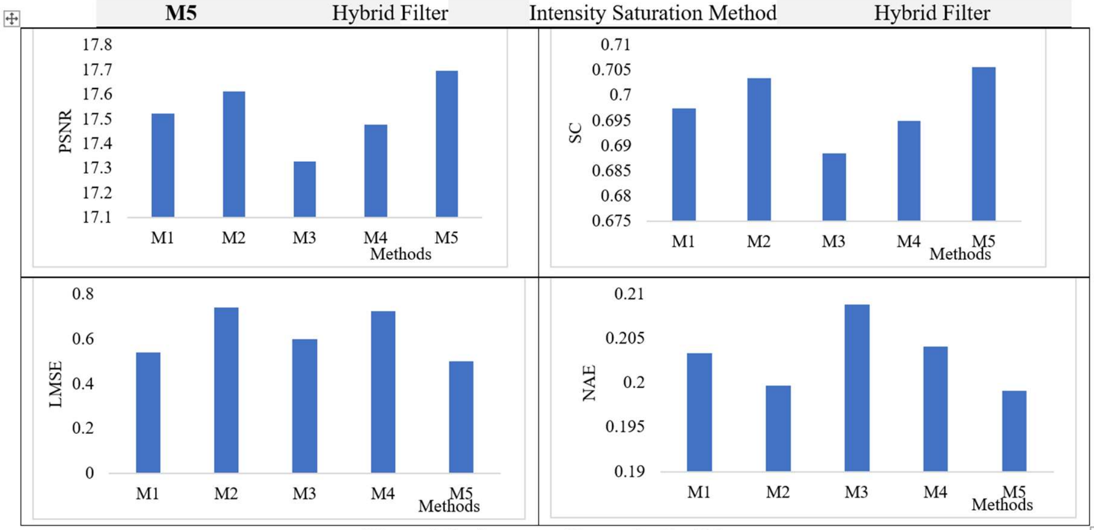


Figure 5: Performance of Image Quality Enhancement

The average performance calculated by measuring each parameter for all the samples provided in the dataset is shown in Figure 5. These findings showed that the M5 outperformed all other combinations of the improvement techniques listed in table 1 in terms of performance. M5 has a greater PSNR, SC, a lower LMSE, and a lower NAE as required by the parameters, which highlights its efficacy. As a result, the M5 approach is used to improve each image sample before being sent to the second phase for additional processing. The suggested system's second phase performs line, word, and character segmentation; the accuracy parameter is computed to analyze the effectiveness of each segmentation level. Accuracy is described as follows:

$$\text{Accuracy}=(T\_inp-C\_out)/T\_inp * 100 \quad (1)$$

Here, T_inp stands for the total number of inputs, which corresponds to the system's total number of Lines, Words, and Characters, and C_out stands for the Correctly Segmented output, also known as Lines, Words, and Characters. The output images that were appropriately segmented are manually reviewed and counted to determine the results. According to the findings, a few samples in each segmentation phase continue to have the issue of under-segmentation. Figure 6 displays an example of under-segmented words, lines, and characters.

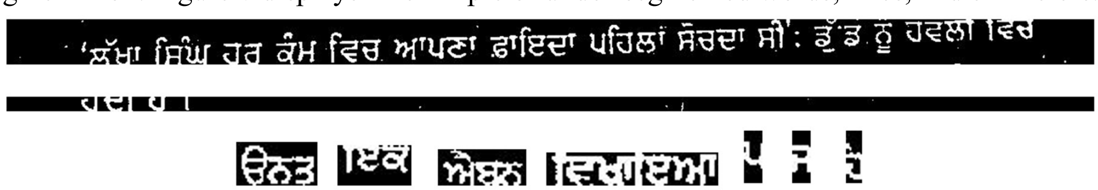


Figure 6: Under-segmented Lines, Words, and Characters

Line Segmentation: By calculating the proper segmentation rate, under-segmentation rate, and over-segmentation rate, the upgraded typewritten dataset images are utilized to assess the effectiveness of the suggested line segmentation approach. Two groups of the 1000 sample dataset, each with between 15000 and 16000 lines, are created. The evaluation's results include the following:

Table 2: Experimentation Results (Line Segmentation)

Dataset	Number of Images	Number of Lines	Correctly Segmented Lines	Under Segmented Lines	Over Segmented Lines
Dataset-1	500	15000	13733	1267	0
Dataset-2	500	16000	14685	1315	0

Word Segmentation: The effectiveness of the suggested word segmentation strategy is evaluated on the line that was recovered from the first phase, also known as line segmentation. The performance in this instance is assessed using the rates of correct, under, and over-segmentation. The line pictures utilized in this experiment are taken from the line segmentation; we chose 2000 of the extracted lines and divided them into two datasets as shown in table 3. The following are some of the outcomes of the word segmentation findings:
The results of the word segmentation findings include the following:

Table 3: Experimentation Results (Word Segmentation)

Dataset	Number of Images (Lines)	Number of words	Correctly Segmented Word	Under Segmented Words	Over Segmented Words
Dataset-1	1000	7520	7199	321	0
Dataset-2	1000	7950	7572	378	0

Character Segmentation: Using the words gathered from the word segmentation stage of the second phase, the success of the suggested character segmentation technique is evaluated. The performance in this instance is assessed using the rates of correct, under, and over-segmentation. Two 1000-word-per-set datasets are taken from the extracted words for this experiment. The evaluation's results include the following:

Table 4: Experimentation Results (Character Segmentation)

Dataset	Number of Images (Words)	Number of characters	Correctly Segmented Characters	Under Segmented Characters	Over Segmented Characters

Dataset-1	1000	3251	2956	295	0
Dataset-2	1000	4208	3893	315	0

Based on the results in tables 3-5, the accuracy for each segmentation level is calculated. Figure 7 compares the accuracy of the AutoSeg technique for line, word, and character.

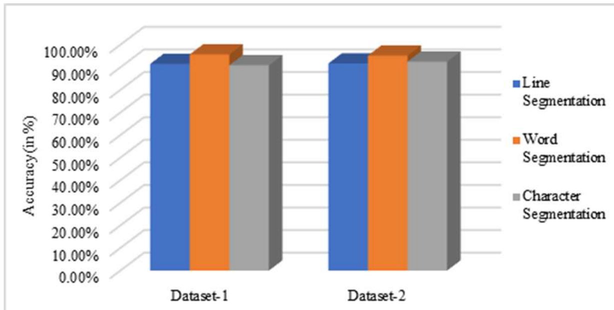


Figure 7: Accuracy of AugSeg Approach

According to the accuracy chart above, word segmentation achieves accuracy rates of 95.7% and 95.2% for datasets 1 and 2, respectively, while line segmentation gets accuracy rates of 91.5% for dataset 1 and 91.7% for dataset 2. However, the final character segmentation achieves an accuracy of 90.9% for dataset 1 and 92.5% for dataset 2, respectively. As a result, the proposed AutoSeg technique, which will be useful for character identification and recognition systems of Gurmukhi text, obtained more than 90% accuracy.

CONCLUSION AND FUTURE SCOPE

Using a typewritten Gurmukhi text recognition system, this work proposes an automated segmentation approach. Additionally, it creates a system that first enhances the image sample's quality before applying the suggested method for automated segmentation. In order to improve the quality of the sample image, noise removal, contrast enhancement, and smoothing were implemented. The combination of the hybrid filter, specifically the wiener and anisotropic, intensity saturation, and hybrid filter, specifically the median and average, functioned well. In order to evaluate the effectiveness of the image quality enhancement phase, PSNR, SC, LMSE, and NAE measurements are used. To further enhance the quality of the segmentation, this augmented image is binarized using the OTSU binarization technique. As a final step, the suggested AutoSeg technique automatically separates the lines while also extracting the words from every line as well as characters from every word. The dataset of 1000 image samples with typewritten text in the Gurmukhi language, separated into two datasets with 500 images each, is used to assess the performance of the segmentation approach. Results show that for both datasets, line, word, and character segmentation accuracy is over 90%, which is highly effective for an automated approach. It also addresses the issue of broken and touching characters. The accuracy of the AutoSeg technique can be tested in the future on printed or handwritten data samples of Gurmukhi text.

REFERENCES

[1] S. D. Pande et al., “Digitization of handwritten Devanagari text using CNN transfer learning – A better customer service support,” Neurosci. Informatics, vol. 2, no. 3, p. 100016, Sep. 2022, doi: 10.1016/j.neuri.2021.100016.

[2] T. Kumar, M. Mahrishi, and G. Meena, “A Comprehensive Review of Recent Automatic Speech Summarization and Keyword Identification Techniques,” 2022, pp. 111–126.

[3] P. Siemund, Multilingual Development. Cambridge. 2022.

[4] S. Srivastava, A. Verma, and S. Sharma, “Optical Character Recognition Techniques: A Review,” in 2022 IEEE International Students’ Conference on Electrical, Electronics and Computer Science (SCEECS), Feb. 2022, pp. 1–6, doi: 10.1109/SCEECS54111.2022.9740911.

[5] P. K. Sarangi, A. K. Sahoo, G. Kaur, S. R. Nayak, and A. K. Bhoi, “Gurmukhi Numerals Recognition Using ANN,” 2022, pp. 377–386.

[6] A. Kumar, S. Bhatiya, M. R. Khosravi, A. Mashat, and P. Agarwal, “Semantic and Context understanding for sentiment analysis in Hindi Handwritten Character Recognition Using a Multiresolution Technique,” ACM Trans. Asian Low-Resource Lang. Inf. Process., Oct. 2022, doi: 10.1145/3557895.

[7] H. Stančić and Ž. Trbušić, “Optimisation of archival processes involving digitisation of typewritten documents,” Aslib J. Inf. Manag., vol. 72, no. 4, pp. 545–559, Jul. 2020, doi: 10.1108/AJIM-11-2019-0326.

[8] A. C. Rouhou, M. Dhiaf, Y. Kessentini, and S. Ben Salem, “Transformer-based approach for joint handwriting and named entity recognition in historical document,” Pattern Recognit. Lett., vol. 155, pp. 128–134, Mar. 2022, doi: 10.1016/j.patrec.2021.11.010.

[9] A. S, G. Z. V, H. M, N. Narayanan, and A. John, “Denoising Heterogeneous Malayalam Document Images Through Binarization,” in 2021 IEEE Madras Section Conference (MASCON), Aug. 2021, pp. 1–6, doi: 10.1109/MASCON51689.2021.9563591.

[10] M. G. Kirschenbaum, Track Changes. Harvard University Press, 2016.

[11] A. Fateh, M. Rezvani, A. Tajary, and M. Fateh, “Persian printed text line detection based on font size,” Multimed. Tools Appl., Jun. 2022, doi: 10.1007/s11042-022-13243-x.

[12] G. Kaur and T. Garg, “Machine Learning for Optical Character Recognition System,” in Machine Vision Inspection Systems, Volume 2, Wiley, 2021, pp. 91–107.

[13] S. Abbas et al., “Convolutional Neural Network Based Intelligent Handwritten Document Recognition,” Comput. Mater. Contin., vol. 70, no. 3, pp. 4563–4581, 2022, doi: 10.32604/cmc.2022.021102.

[14] Z. Fu, H. Xie, S. Fang, Y. Wang, M. Xing, and Y. Zhang, “Learning Pixel Affinity Pyramid for Arbitrary-Shaped Text Detection,” ACM Trans. Multimed. Comput. Commun. Appl., Jul. 2022, doi: 10.1145/3524617.

[15] G. Kaur and A. Kumar, “Automated Line Segmentation for Gurmukhi Text Recognition System,” in 2021 2nd International Conference on Computational Methods in Science & Technology (ICCMST), Dec. 2021, pp. 264–269, doi: 10.1109/ICCMST54943.2021.00062.

[16] Y. Zhong, X. Cheng, T. Chen, J. Zhang, Z. Zhou, and G. Huang, “PRPN: Progressive region prediction network for natural scene text detection,” Knowledge-Based Syst., vol. 236, p. 107767, Jan. 2022, doi: 10.1016/j.knosys.2021.107767.

[17] R. P. Kaur, M. K. Jindal, and M. Kumar, “Text and graphics segmentation of newspapers printed in Gurmukhi script: a hybrid approach,” Vis. Comput., vol. 37, no. 7, pp. 1637–1659, Jul. 2021, doi: 10.1007/s00371-020-01927-0.

[18] R. Goyal, D. R. K. Narula, and D. M. Kumar, “A Novel approach for Segmentation of Typewritten Gurmukhi Script,” Int. Journalof Adv. Sci. Technol., vol. 27, no. 1, pp. 77–96, 2019.
[19] F. Slimane, A. Mazzei, O. Topalov, G. Verzi, and F. Kaplan, “A web-based tool for segmentation and automatic transcription of historical documents,” in 2017 International Joint Conference on Neural Networks (IJCNN), May 2017, pp. 2730–2737, doi: 10.1109/IJCNN.2017.7966192.
[20] E. Paulus, M. Suryani, and S. Hadi, “Improved Line Segmentation Framework for Sundanese Old Manuscripts,” J. Phys. Conf. Ser., vol. 978, p. 012001, Mar. 2018, doi: 10.1088/1742-6596/978/1/012001.
[21] S. R. Narang, M. K. Jindal, and M. Kumar, “Line Segmentation of Devanagari Ancient Manuscripts,” Proc. Natl. Acad. Sci. India Sect. A Phys. Sci., vol. 90, no. 4, pp. 717–724, Oct. 2020, doi: 10.1007/s40010-019-00627-2.
[22] B. K. Barakat et al., “Unsupervised deep learning for text line segmentation,” in 2020 25th International Conference on Pattern Recognition (ICPR), Jan. 2021, pp. 2304–2311, doi: 10.1109/ICPR48806.2021.9413308.
[23] D. Dahake, R. K. Sharma, and H. Singh, “On segmentation of words from online handwritten Gurmukhi sentences,” in 2017 2nd International Conference on Man and Machine Interfacing (MAMI), Dec. 2017, pp. 1–6, doi: 10.1109/MAMI.2017.8307870.
[24] M. K. Mahto, K. Bhatia, and R. K. Sharma, “Segmentation of Offline Handwritten Gurmukhi Words Using Projection Features,” in 2019 8th International Conference System Modeling and Advancement in Research Trends (SMART), Nov. 2019, pp. 230–234, doi: 10.1109/SMART46866.2019.9117480.
[25] P. Sahare and S. B. Dhok, “Robust Character Segmentation and Recognition Schemes for Multilingual Indian Document Images,” IETE Tech. Rev., vol. 36, no. 2, pp. 209–222, Mar. 2019, doi: 10.1080/02564602.2018.1450649.
[26] P. A. Tamhankar, K. D. Masalkar, and S. R. Kolhe, “A Novel Approach for Character Segmentation of Offline Handwritten Marathi Documents written in MODI Script,” Procedia Comput. Sci., vol. 171, pp. 179–187, 2020, doi: 10.1016/j.procs.2020.04.019.
[27] B. Thakral and M. Kumar, “Devanagari handwritten text segmentation for overlapping and conjunct characters- A proficient technique,” in Proceedings of 3rd International Conference on Reliability, Infocom Technologies and Optimization, Oct. 2014, pp. 1–4, doi: 10.1109/ICRITO.2014.7014746.
[28] M. Li, M. Bai, and Y. Lv, “Text segmentation by integrating hybrid strategy and non-text filtering,” Multimed. Tools Appl., Jun. 2022, doi: 10.1007/s11042-022-13029-1.
[29] O. Elharrouss, S. Al-Maadeed, J. M. Alja’am, and A. Hassaine, “A Robust Method for Text, Line, and Word Segmentation for Historical Arabic Manuscripts,” in Data Analytics for Cultural Heritage, Cham: Springer International Publishing, 2021, pp. 147–172.
[30] M. Ayesh, K. Mohammad, A. Qaroush, S. Agaian, and M. Washha, “A Robust Line Segmentation Algorithm for Arabic Printed Text with Diacritics,” Electron. Imaging, vol. 2017, no. 13, pp. 42–47, Jan. 2017, doi: 10.2352/ISSN.2470-1173.2017.13.IPAS-204.
[31] Y. Loijing Khomba Khuman, H. Mamata Devi, and N. Ajith Singh, “Entropy-based skew detection and correction for printed Meitei/Meetei script OCR system,” Mater. Today Proc., vol. 37, pp. 2666–2669, 2021, doi: 10.1016/j.matpr.2020.08.522.

[32] M. Sravani, A. Maheswararao, and M. K. Murthy, "Robust detection of video text using an efficient hybrid method via key frame extraction and text localization," Multimed. Tools Appl., vol. 80, no. 6, pp. 9671–9686, Mar. 2021, doi: 10.1007/s11042-020-10113-2.

[33] P. Hu, W. Wang, Q. Li, and T. Wang, "Touching text line segmentation combined local baseline and connected component for Uchen Tibetan historical documents," Inf. Process. Manag., vol. 58, no. 6, p. 102689, Nov. 2021, doi: 10.1016/j.ipm.2021.102689.

[34] R. Islam, M. R. Islam, and K. H. Talukder, "An efficient ROI detection algorithm for Bangla text extraction and recognition from natural scene images," J. King Saud Univ. - Comput. Inf. Sci., Mar. 2022, doi: 10.1016/j.jksuci.2022.02.001.

[35] V. Göreke, "A novel method based on Wiener filter for denoising Poisson noise from medical X-Ray images," Biomed. Signal Process. Control, vol. 79, p. 104031, Jan. 2023, doi: 10.1016/j.bspc.2022.104031.

[36] J. Yang, Z. Guo, D. Zhang, B. Wu, and S. Du, "An anisotropic diffusion system with nonlinear time-delay structure tensor for image enhancement and segmentation," Comput. Math. with Appl., vol. 107, pp. 29–44, Feb. 2022, doi: 10.1016/j.camwa.2021.12.005.

[37] P. Singh, A. K. Bhandari, and R. Kumar, "Naturalness balance contrast enhancement using adaptive gamma with cumulative histogram and median filtering," Optik (Stuttg)., vol. 251, p. 168251, Feb. 2022, doi: 10.1016/j.ijleo.2021.168251.

[38] J. Zhou, X. Wei, J. Shi, W. Chu, and W. Zhang, "Underwater image enhancement method with light scattering characteristics," Comput. Electr. Eng., vol. 100, p. 107898, May 2022, doi: 10.1016/j.compeleceng.2022.107898.

[39] B. Karthick, H. Singh, and M. Malarvel, "Pre-processing Techniques for Offline Tamil Handwritten Character Recognition," in 2022 7th International Conference on Communication and Electronics Systems (ICCES), Jun. 2022, pp. 976–981, doi: 10.1109/ICCES54183.2022.9835777.

[40] B. Rocha et al., "Skew Angle Detection and Correction in Text Images Using RGB Gradient," 2022, pp. 249–262.

[41] R. Remya, B. P. Shan, K. Umamaheshwari, D. J. Derwin, and D. Lavanya, "Improved DWT Algorithm for Filtering of MRI Images for an Efficient Diagnosis," in 2022 Second International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), Apr. 2022, pp. 1–8, doi: 10.1109/ICAECT54875.2022.9807955.