# Post-Processing Techniques To Enhance The Performance Of Gurmukhi Text Recognition System

**Dr. Sukhdev Singh**

Assistant Professor, Multani Mal Modi College, Patiala.

**Abstract:** The development of the Gurmukhi Text recognition system is a challenging task as Gurmukhi text characters tend to intermix with other characters because of inherited structure. Particularly when the machine is trained to extract and recognize Gurmukhi text from natural science, the problem turns into many folds. The natural scene images carry complex backgrounds which intermixed with text regions and raised various issues in recognition of text. The present piece of work discusses experiments carried out to improve the performance of the system developed for the extraction of Gurmukhi text from natural scene images. The study revealed that dictionary-based error detection and correction is the best method to improve the results of the system. Using post-processing, the present system's performance level raised from 92.0% to 94.5% for Gurmukhi alphabets and 95.5 to 98.5% for digits.

**Keywords:** Gurmukhi text recognition, Post-processing, Error Detection, Dictionary Lookup table search.

## 1. Introduction

Text extraction from natural scene images is a difficult task that is comparably complex to text extraction from typical scanned document systems (OCRs). The complexity of the system is influenced by some things. The most efficient way to start up the performance of such a system is through post-processing procedures. The performance of the text extraction system is improved by the use of post-processing techniques. It is a crucial component of any OCR. The extraction of Gurmukhi text from photographs of natural scenes follows the same procedure. The following post-processing techniques are implemented:

i. Error detection
ii. correction in character recognition.

The post-processing is applied immediately after the recognition process where the output of the recognition process acts as input for the post-processing phase. The purpose of the post-processing techniques is to validate recognized words or Suggested alternative words. The text present in the natural scene image is closer to the printing text which makes it easy to recognize. But various issues are involved in the recognition process. These issues are as follows:

- Text in Gurmukhi script is split into three zones namely upper zone, middle zone and lower zone. The headline connects upper characters with middle zone characters. The headline is required to be removed so that upper and middle sub-characters can be segmented. It has happened sometimes that some regions of a sub-character image that touches upper zone may be stripped off while separating both zones. For example, .

After trimming the headline some of the Gurmukhi characters generate similar character features which are difficult to distinguish. Table 1 shows similar characters' features which are difficult to distinguish.

Table1: Removal of headline cause confusion while recognizing the character

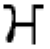
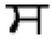
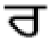
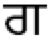
| | | | |
|---|---|---|---|
|  | |  | |
| Fig.(a) | Fig.(b) | Fig.(c) | Fig.(d) |
|  | |  | |
| Fig.(e) | Fig.(f) | Fig.(g) | Fig.(h |

Fig (a) and Fig (c) are with headlines and when these headlines are removed from that result in similar images, as shown in Fig (b) and Fig (d). Similarly, Fig (e) and Fig (g) are with headlines and when these headlines are removed from that result to similar images, as shown in Fig (h), and Fig (h).

To extract different features from Gurmukhi characters, binarization and thin are utilized. A few Gurmukhi characters have identical skeletons, which makes it difficult to recognize them. Table 2 below provides a list of characters with comparable skeletons.

Table2: Example of Gurmukhi characters having similar skeletons after

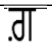
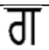
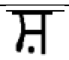
| | | | |
|---|---|---|---|
| | |  |  |
| Fig.(a) | Fig.(b) | Fig.(c) | Fig.(d) |
| | |  | |
| Fig.(e) | Fig.(f) | Fig.(g) | Fig.(h) |

Fig. (a) and Fig. (b) have similar skeletons characters with a very minor difference at the bottom of the character (Lower zone). Fig.(c) and Fig (b) have a difference of lower sub-character only thinning and binarization. Fig. (e) and Fig. (f) have similar skeletons as there is the only difference of lower characters. Fig. (g) and Fig. (h) have only differences in the middle zone.

The following list of post-processing methods is taken from the literature: contextual thinking. Dictionary matching or word validation technique

- Shape similarity-based processing
- N-grams based validation
- Language Translation

The word validation technique based on the dictionary lookup method is best suitable for word-level post-processing methods. In the methods, the recognized word is searched in the pre-existing databases of words. If the word is found in the database then the recognized word is accepted otherwise a list of recommended words based on some similarity rules is listed. The problem with this method is the storage and searching of huge word databases. To overcome the problem of large databases and long searching time, the concept of N-gram is introduced. The N-gram is a technique to store data in string format such that it can represent the different combinations of characters.

The word database is saved as strings that represent various word combinations. Another strategy uses the word collection table to store groups of related terms to improve the efficiency of searches. An OCR error detection and correction method for Bangla has been developed by Chanda [2]. the method used to divide input words into different sections, such as suffixes and root lexicons. Using lexicons containing root and suffix pairs, the input words are checked for grammatical agreement. It is a challenging situation to identify the words which have structural similarities for example: ਚੋਰ (CHOR) or ਚੌਰ (CHAUR), ਸੇਵਾ (SEVA) or ਮੇਵਾ (MEVA). To handle such problems system has implemented consonants based special encoding technique. The Gurumukhi text recognition returns a sequence of characters to be verified and replaced with the correct one. The classification phase does not always provide accurate results. The results need to be refined using post-processing techniques. Error detection and correction is one of the most commonly used methods to enhance the performance of the recognition system.

The Gurmukhi script's eight subsets of alphabets are classified according to the similarity of shape in the current scheme for mistake detection and correction. The closeness of shape rules makes it easier to group alphabets into subsets that have similar geometry. As an illustration, the Gurmukhi characters are similar but differ in their upper headline. Therefore, if either of these two characters is misspelled, the appropriate replacement should be chosen among these two. Similarly , a list of such combinations will aid in generating the best possible adjustment options.

For example, Gurmukhi characters ਮ and ਸ, are similar but they have a difference in the upper headline. So, if any one character out of these two is misspelled then alternative should be from these two. Similarly, a list of such combinations will help to generate the most appropriate correction alternatives.

- Initially, The Gurmukhi script is classified into a character set of 8 subsets based on their shape similarities.

For example, sub-code 1 represents ੳ, 2 represents ਚ, ਟ, ਰ, ਹ, ਢ, ਦ, ਫ, ਫ਼, ੲ, ਝ, ਵ, ਛ, similarly other characters are coded.

- The second step was creating the database dictionary, which contains 74,205 words such as street names, city names, and words that are often used in banners and notice boards. The third step involves consonant-specific special codes for validation.

- The third stage is matching each word from the reorganization phase that was detected with an AVL tree-stored dictionary consonant. The source word may be used to correctly recognize the term if it appears in the word list of the word code; if not, a highly-ranking word from the list will be proposed.

### Search word using Dictionary Lookup Method:

The word is looked up in the AVL tree, where the left and right values, respectively, stand in for the left and right subtrees. The height of the right subtree is subtracted from the height of the left subtree to calculate the balancing factor (BF), which is used to keep the tree balanced. To search for the word in the dictionary, use the following process.

**Check the word in the dictionary.**

a) If the word is present in the dictionary that means that word is correct and no change is required there.

b) If the word is not found in the dictionary, and then the word might be misspelled.

**Move to step 2.**

An algorithm is run to produce a list of replacement possibilities for the typo.

a) The process is carried out on the list of created word ideas.

b) Highest ranked word is suggested and replaced with the misspelled word.

### Results and Discussions

The recognition rate has increased by an average of 2.5% for Gurmukhi alphabets and 3.0% for digits thanks to error detection and rectification. Table 3 displays the rate of recognition improvement.

Table3: Progress in performance of the system after Post processing

| **Fields** | **Before** | **After** | **Improvement** |
|---|---|---|---|
| Alphabets | 92 | 94.5 | 2.5 |
| Digits | 95.5 | 98.5 | 3.0 |

The overall result of recognition has improved up to 95.3 % for machine-printed Gurmukhi text extracted from Natural scene images.

**Conclusion:** Text detection from natural scene images is a prominent area of research in the field of computer vision and can be utilized to develop a wide range of applications. The performance of such a system can be enhanced by supporting post-processing techniques. The present research has utilized post-processing techniques to enhance the performance of the system developed for the detection and recognition of Gurmukhi text from natural scene images. The system incorporated with dictionary lookup method and enhance the performance of the system from 92% to 94.5 % (Gurmukhi Alphabets) and 95.5% to 98.0%(Digits).

**References:**

[1]. Chanda, Sukalpa, Katrin Franke, and Umapada Pal. "Structural handwritten and machine print classification for sparse content and arbitrary oriented document fragments." In Proceedings of the 2010 ACM Symposium on Applied Computing, pp. 18-22. 2010.

[2]. Chowdhury, Mohammad Isbat Sakib, Barnali Dey, and Md Saifur Rahman. "Segmentation of printed Bangla characters using structural properties of Bangla script." In 2008 International Conference on Electrical and Computer Engineering, pp. 639-643. IEEE, 2008.

[3]. S.V. Rajashekararadhya, P. Vanaja Ranjan," Efficient zone based feature extraction algorithm for handwritten numeral recognition of four popular south Indian scripts", Journal of Theoretical and Applied Information Technology.1171-1180. 2008.

[4]. Sinha, R. M. K. "Rule based contextual post-processing for Devanagari text recognition." Pattern Recognition 20, no. 5 (1987): 475-485.

[5]. Sopharak, Akara, Bunyarit Uyyanonvara, Sarah Barman, and Thomas H. Williamson. "Automatic detection of diabetic retinopathy exudates from non-dilated retinal images using mathematical morphology methods." Computerized medical imaging and graphics 32, no. 8 (2008): 720-727.

[6]. Takahashi, Hiroyasu, Nobuyasu Itoh, Tomio Amano, and Akio Yamashita. "A spelling correction method and its application to an OCR system." Pattern Recognition 23, no. 3-4 (1990): 363-377.

[7]. Rong, Li, En MengYi, and H. Zhang. "Object Proposals for Text Extraction in Natural Scenes using Ensemble RankSVM." In 12th International Workshop on Document Analysis System (DAS). 2016.