Segmenting Touching Characters in Nepali OCR

Sanjeev Maharjan

Abstract:

Effective segmentation of the characters, yield the better character recognition results for any OCRs. Nepali script being more or less similar to Devnagari scripts, which constitutes numerous touching compound characters. This report presents the significance of the touching characters in Nepali Scripts and how these touching characters are causing the problems. This report also presents implementation of the technique for segmenting these touching characters and improving the efficiency of Nepali OCR, named as Multi-Factorial Analysis.

1. INTRODUCTION

Optical Character Recognition is process of recognizing the text from the scanned image of text and converts such recognized text into machine editable format. Following are the steps that are conducted during the process of OCR:

1. Scanning of the text document
2. Pre-processing the document
3. Segmentation
    - 3.1. Line Segmentation
    - 3.2. Word Segmentation
    - 3.3. Character Segmentation
4. Post-Processing
5. Character Recognition
6. Preparing the editable text

These steps are common among all the OCRs that exist for different languages. The only complexity that would arise for different languages is how to implement these steps. The method for segmentation is the crucial among the East-Asian languages like Devnagari, Nepali, Bangla etc.

Since above mentioned Asian Scripts possess the words connected by 'ShiroRekha', the character classifier which normally functions as per the connected components, might face difficulty to segment such words into characters. But, rather than the headline, the existence of large number compound characters (formed by two or more basic characters) in these scripts is the area of concern as not all of these compound characters can be trained.

Besides the compound characters, there may exist other touching characters due to poor scanning and due to the scanning of old documents. This is the cause of efficiency degradation for not only East-Asian Scripts but for all different scripts.

2. LEARNING ABOUT NEPALI SCRIPT

Like the other East-Asian Scripts such as Devnagari, Nepali Scripts have horizontal writing style, i.e. form left to right and have no upper and lower case distinction.

Nepali script is pretty similar to that of Devnagari, and that it also possesses the basic character set with vowels and consonants, and that vowels act as modifiers when combined with consonants.

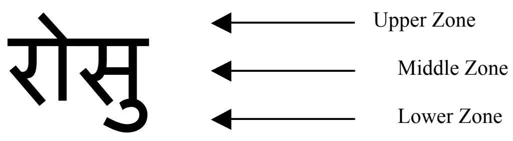
Eg:
Vowels:
अ आ ओ औ इ उ
Consonants:
क ख ग घ
Modifiers:
◌ि ◌ी ◌ू ◌ु ◌ा
Modifiers with consonants:
कि की कू कु का

Nepali scripts do possess a dark line that connects all the words, which we call them as

‘Shirorekha’ or headline. Nepali scripts can be divided into three zones:

- **Upper zone**: portion above headline with upper modifiers
- **Middle zone**: portion with character
- **Lower zone**: potion below character with lower modifiers

Eg:



3. TOUCHING (COMPOUND) CHARACTERS IN NEPALI SCRIPT

As specified above, Nepali Scripts do consists of compound characters which are formed by combining two or more basic characters. Following figure will illustrate few such compound characters:

क्व ख्व ग्व घ्व ङ्व च्व छ्व ज्व झ्व ञ्व
ट्व ठ्व ड्व ढ्व ण्व त्व थ्व द्व ध्व न्व
प्व फ्व ब्व भ्व म्व य्व र्व ल्व व्व स्व

FIG: Compound Characters in Nepali Script

These compound characters as shown in the above figure constitute of two or more basic characters, and take an entirely different shapes and forms when formed.

Due to these compound characters, the Nepali Scripts possess whole new complexity for the OCR implementation as these compound characters are touching characters. Since there is no limitation in the formation of the compound characters, not all compound characters can be trained for such OCRs.

Alternative to the training of all possible compound character set, is to segment these touching compound characters into two separate characters, as resultant of such segmentation of touching characters can be easily trained, as they are limited by the number of the basic character set.

4. SEGEMENTATION COMPOUND NEPALI CHARACTERS

Though there do exist various approaches to segment such compound characters, I have adopted the technique of Multi-Factorial Analysis. This technique was explained in the paper entitled, “Segmentation of Touching Characters in Printed Devanagari and Bangla Scripts Using Fuzzy Multi-Factorial Analysis”.

The reason behind selecting this technique, is the efficiency and accuracy result, that has been observed for Devnagari and Bangla Scripts which is about 98%. Since Nepali script resembles to Devnagari with only few modifications, this technique appeared to be fruitful to segment compound Nepali characters as well.

4.1 What is Multi-Factorial Analysis?

In 1982 Wang first defined the concept of factor spaces and applied it to the study of Artificial Intelligence. According to him, ‘*factor*’ is a primary term with properties like *state* and *characteristics*.

H.X. Li and V.C. Yen have discussed four types of factors:

1. Measurable Factors: (like time, height etc)
2. Nominal Factors: (like religion)
3. Degree Factors/ Fuzzy Factors (Degree of Similarity, Feasibility)
4. Switch or Boolean Factors only two possible values

The relevant factors for the touching characters are fuzzy factors and to segment the touching characters multiple fuzzy factors are considered to identify the optimal cut columns.

4.2 Factors to Consider for Compound Character Segmentation

Till date the following factors are considered and evaluated, for the multi-factorial analysis:

1. Inverse Crossing count
2. Measure of Blob Thickness
3. Degree of Middle-ness

Inverse Crossing Count (Fic) is calculated to check the number of black pixel to white pixel transaction during a column scan. If such transition count is greater than 1 then that column is less favorable for being a cut column for segmentation. Mathematically,

Fic= c^{-1}
Where c is the vertical crossing count for a pixel column

Measure of Blob-Thickness (Fmt) is calculated to check the number of black pixels encountered during a column scan. Column with smaller number of such pixel count are more preferable for being the cut column.
Mathematically,

Fmt=1-t/T
Where
t= no. of black pixels in 1 column scan,
T=height of the characters middle zone

Degree of Middle-ness (Fdm) is the factor calculated to check where the blob lies in the middle zone of the character. If the blob is more or less at the middle position of the middle zone then such column can be preferably cut column. Mathematically,

Fdm= min(l1,l2) / max(l1,l2)
Where, l1 is the distance of the blob from the headline, and l2 is the distance of the blob from the end of middle zone

4.3 Performing Multi-Factorial Analysis

Once the above factors are calculated, then the multi-factorial analysis is to be performed. If an image of touching characters has **m** as the total number of the pixel column, then for all the m columns above three factors are evaluated and a 3X**m** one–factor evaluation matrix is formed as follows:

$$V_s = \begin{bmatrix} f_{ic}^1 & f_{ic}^2 & \cdots & f_{ic}^m \\ f_{mt}^1 & f_{mt}^2 & \cdots & f_{mt}^m \\ f_{dm}^1 & f_{dm}^2 & \cdots & f_{dm}^m \end{bmatrix}.$$

Each column in matrix represents each pixel column in the image and consists of a 3-D vector that reflects three different aspects for that column to be a cut column. Next, these three aspects get combined and mapped into a 1-D scalar by an ASM function M_s. The function M_s transforms the 3xm matrix Vs into a 1xm matrix V's as follows:

$$V_s' = (f_s^1, f_s^2, \ldots, f_s^m)$$
$$\text{where } f_s^i = M_s(f_{ic}^i, f_{mt}^i, f_{dm}^i)$$

5. RESULTS FROM THE MULTI-FACTORIAL ANALYSIS

Following are the results from the segmentation of the sample compound character, after performing the multi-factorial analysis and identifying the cut position. These cut positions are selected from the 1Xm matrix evaluated above where column with highest value is selected first and then second highest is selected and so on.

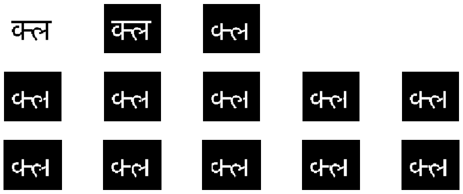


FIG: Output of Three Factor Analysis

The above implementation resulted a two valid basic characters at 7th iteration of cut-column verification.

6. CONCLUSION

Though Nepali script does possesses due to the existence of touching characters, the segmentation of such touching points is preferable for improving the performance of any OCR.

Hence, the result of multi-factorial analysis is looking very promising, and would only get better with inclusion of more factors such as upper formed at above and below the touching point.