OCR Software for Printed Tamil Text

Dr V. Krishnamoorthy <profvk@softhome.net>

Professor, Crescent Engineering College (Former Professor Of Anna University)
11, Fourth Street, Padmanabha Nagar, Adayar, Chennai 600 020

Abstract

In this paper we discuss the problems faced when trail runs were made to use the Tamil OCR software. For neatly printed texts we could get more than 99% accuracy. But many wanted to use the software for old books which are printed mode than a few decades. In these books it is found that the printing quality is poor in many cases. Hence the percentage of accuracy came down depending on the quality of the printing. The main objective in this scenario is to reduce the overall conversion time. The enhancements, solutions provided and the possible enhancements and limitations of the OCR software are discussed in this paper.

Introduction

When the software for printed Tamil text was started, the main aim of the project was to see how the new technology being developed will work. As such, the aim was to recognize letters printed correctly, without any distortion. The first version of the software was demonstrated at the INFITT Conference at Kolalumpur, Malaysia in August 2001. It was shown that the new methodology worked well at high speed. This version had the following limitations.

1. One line should have same font.
2. Normal and italics should not be mixed.
3. One line should have letters of the same size.
4. Letters should not be underlined.
5. Pictures should not be present in a page.
6. More than one column should not be there.
7. Old type of nai and naa should not be present.

Excepting the fourth limitation, all the others have been solved and the software was released in December 2001. When the software was presented to the prospective customers, the following requirements came to light.

Problems

There is a demand for publishing old books. Some of the books were printed more than a hundred years ago. Some such books were scanned, and it was found that in some cases, the printing quality is very poor, and the following problems were noted.

The letters were found broken into disjoint pieces and parts of them were missing. Adjacent letters were merged together. Some letters were smudged to the extent that only the outline is seen and the entire inside is printed black. The lines were not printed in a horizontal fashion. The lines moved up and down. The letters in the same line were of different thickness. The distance between two characters are not uniform. A word gets broken into two parts and put

on adjacent lines. At the left or right sides of the printed area, lot of noise is seen touching the letters. When two small size pages were scanned together, each page had a different orientation, due to uneven binding.

Different publishers use the fonts of different encoding. Many of them are yet to convert to TAM or TAB which are the officially recognized standards in Tamil Nadu. They wanted the output of the OCR to be given in their encoding. Also they want the corrections to be done using the keyboard layout they have been using.

Solutions

Since the letters break in unpredictable fashion and become disjoint, it is difficult to identify the letter. To speed up the total OCR time we have provided for interactive method in recognition phase. When a part could not be recognized, it shows the picture of the current line and the part which could not be identified. The correct character or characters (in the case of many letters joined together to form a single connected unit) can be chosen from a table and given. Also many of the noises can be eliminated at this stage.

At this stage, we can also ask the software to train a particular character using the picture of the unrecognized character. This helps in enriching the training of the software. In the later stages this training will increase the recognition percentage.

Provision has been made for handling characters of different height within the same line. Also hanging indent which spans many lines is taken care of.

Provision has been made to delete unwanted portions. Also only the portions of a page which are to be read can be chosen one after another, in the required order.

Providing the output in different encoding has been provided, to cater to the immediate needs of the publishers. This created another problem. The OCR works on font information file created by training the software for a particular font. This is to enhance the success rate. But when different encoding and different keyboard layouts are to be provided, it poses a big challenge. All have to be accommodated without any problems. For this, the training module has been reorganized thoroughly. This section needs only clicking on the desired characters and eliminates keyboard input. In other places the user can edit the output text using his own keyboard driver.

A spell check has been provided within the OCR software. This had to be modified to cater to the different encoding.

It is found that many a time letters which look very much similar gets recognized wrongly. For example, (la and va), (ka and su) etc. A special module to spell check a recognized word automatically and pick the correct alternative from the above type of pairs is being tried out. Some times both such words may be correct. For example the words kakku and sukku are correct. Except for these rare cases others can be eliminated. If the number of such doubtful letters are many in word then the number of combinations to be checked grow rapidly. All the correct words can be shown in the suggestion list, and the user can just pick the right one. This takes more time. We are trying to see whether we could analyze the situation intelligently and reduce the number of combinations, so that the time taken is not very much.

The problem of two pages being scanned together is solved by rotating the page by 90 degrees first to bring the pages to a horizontal fashion. After that, each page can be adjusted individually, so that the lines are very nearly horizontal, before each page is recognized.

Conclusion

The OCR software with the above modifications has been tested in a real life situation. A small book with about 140 pages has been scanned and converted into the e-form. About 30 pages could be converted per day, using a 400 MHz Celeron computer. This includes scanning, reading, spell checking and doing the corrections on the screen. One proof correction using a printout was carried out. In doing this test we found that the positive aspect is that no line or word will be left out, and no line will be repeated. But there is a possibility that pages can be left out. But this will be found out at the print out proof correction level, based on the continuity.

The elimination of typing and one proof correction will be saving time and cost to an appreciable extent. Use of faster computer will reduce the time considerably. Also a person not conversant with fast Tamil typing can also achieve good throughput.

The spellchecker incorporated in the software is built for modern Tamil. In case the text is of old origin, the spell checking will not be that effective. This has to be borne in mind while using it.

Due to the quality of printing, it is found that broken letters and heavily smudged letters could be seen in some old printed books. It is not possible to machine read such a poor quality printed text with very high accuracy. But the many facilities incorporated in the software, like interactive reading, normal and special spellchecks, reduce the total time for reading a text considerably. This is what could be achieved practically today.
