

Computer Recognition of Off-line Handwritten Kannada Sentences

Abstract

This thesis investigates the combination of a syntax analysis module with an offline recognition system for handwritten Kannada sentences. The goal of this research is machine recognition of offline handwritten Kannada sentences. This is one step towards the ultimate goal of automatic machine reading of complete pages of text. In this work, we have also added the document image mosaicing tool which performs effective mosaicing of handwritten documents. The problem can be decomposed into the recognition of the sequence of words that are embedded into the stroke objects of the sentence image. These objects must be properly grouped to isolate the individual words. This is a difficult problem because of the wide range of handwriting styles and word spacing. One working solution is to build upon the recent work in off-line isolated word recognition. Thus a two step approach to sentence recognition is taken. The task of separating lines and words in the document is fairly independent of the script and hence can be achieved with standard techniques. However, Due to the peculiarities of the Kannada script, a novel segmentation scheme proposed by can be used , where words are first segmented to a sub-character level, the individual pieces are recognized and these are then put together to effect recognition of individual *aksharas* or characters. We have used different types of features for the classifiers. For the classification of these feature vectors we have considered neural network based classifiers such as Feed-forward with Back-propagation (Supervised) and Kohonen Self organizing Map (Unsupervised) and Statistical based method such as Support Vector Machines (SVM).

There are many OCR systems are available for handling printed, handwritten and cursive scripts in English documents. But there are not many OCR systems for Asian Languages. The work reported in this thesis is motivated by the fact that there are no reported efforts at developing document analysis systems for the south Indian Languages. The OCR 's task is to identify the characters of Kannada script and the word processor provides an interface for viewing and editing documents in Kannada. The primary task of the syntactical analyzer is to correct any errors made by the OCR as well as to provide a spell-checking tool to the word processor. The syntactical

analyzer will incorporate a dictionary of known words, built from a simple word list and organized for efficient searching. The controller will read words from a text file, one by one and pass them to the syntactical analyzer for processing. If the syntactical analyzer finds an exact match, it will simply indicate the word has been spelt correctly. If not, the syntactical analyzer will provide the controller with a list of suggested alternatives for the submitted word. To select suggestions, the syntactical analyzer will need some rules for deciding whether a match is close enough.

The Post-processing part of the system is the most problematic part of the system. Post-processing models are unavoidable for applications such as phrase recognition and sentence recognition. Handwriting recognizers output a list of word choices with their scores for each input word position. The use of natural language knowledge is to filter confusion cases from the recognition results and improve sentence level recognition result.

In this work, the sequence of operations carried out is as follows. A page of handwritten Kannada text is scanned through a flat bed scanner at 300dpi. The image format used is the bmp format. All the preprocessing steps (like skew detection and correction, slant removal etc.) are applied for the input image. We have implemented various skew detection algorithms, such as Hough Transform, Wigner-Ville Distribution, standard deviation and variance based methods. The skew and slant corrected image is then input into the segmentation program which separates lines, words and then segment words into smaller parts. In the final step we need to label each of the segments using a classifier and then affect final recognition of the aksharas based on these labels. Before presenting the final accuracies obtained on some data with the system, we experimented with different types of features vectors. The first feature set based on (No. of holes, pixel distribution, moments, asymmetry, etc), the second one based on dividing image into radial track and sectors and the third with Zernike Moments as features. We have used support vector machines (SVM) and neural network based classifiers for recognition. The word processor accepts a word from the OCR module for word recognition. This word may have one or more characters which cannot be identified with certainty. The word processor also accepts a list of probable substitutions for the characters along with the probabilities of those substitutions being the right substitutions from the OCR module. The word processor

searches the lexicon for possible corrections to the word. This is done by comparing the word with previously identified complete words which are in the dictionary. A list of probable words is compiled along with the probability of each of them being the required corrected word. The word processor outputs a list of suggested corrections based on the results of a search and the frequency of occurrence to the word processor. The data structure used for the word processor is the ternary search tree. Ternary trees can store a huge amount of information in little space and the retrieval time is small. Ternary search trees combine the best of two worlds, the low space overhead of binary search trees and the character-based time efficiency of digital search trees. Sentences are parsed to list the grammatical classes of each incorrect candidates then lexical query searches for words in a lexicon according to grammatical classes.

In the next step each word of the sentence is given as input to the morphological analyzer. The morphological analyzer does the analysis of each word and gives the category, gender etc., for each word. The output of the morphological analyzer undergoes POS disambiguation. The part of speech (POS) tagged text is submitted to the parser. Now the parser can automatically use the probabilities assigned to the various productions to find the most probable sentence. The parsed sentences are then returned to the user in non increasing order of probabilities of their correctness.

Under the guidance of

Dr.S.C.Sharma

Professor & Director

CMRTU, R&D

R.V.College of Engg., Campus

Bangalore – 59

Ramankanthkumar P
Research Scholar, CMRTU,
R&D,R.V.College of Engg. Campus
Bangalore – 59

* * * * *