Development of an OCR System

Nathan Harmata

TJHSST Computer Systems Lab 2007 - 2008

Abstract

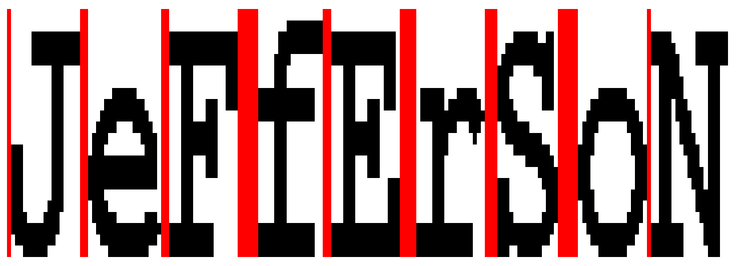
OCR (Optical Character Recognition) is a very practical field of Computer Science. Since the late 1980's, researchers have been developing system to identify text from non electronic sources, such as pictures or newspapers. The use of OCR systems has spanned from making books in Braille to sorting mail by zip code.

Background

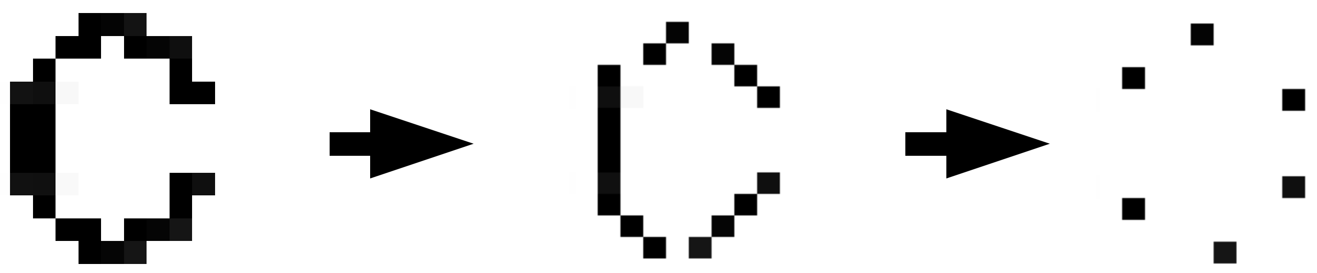
Although there are a few options currently available to the public, like Microsoft Document Imaging, most of them are either unused or not accurate enough. The goal of this project is to create an OCR system that is simple to use and can handle most formatting and fonts.

Procedures

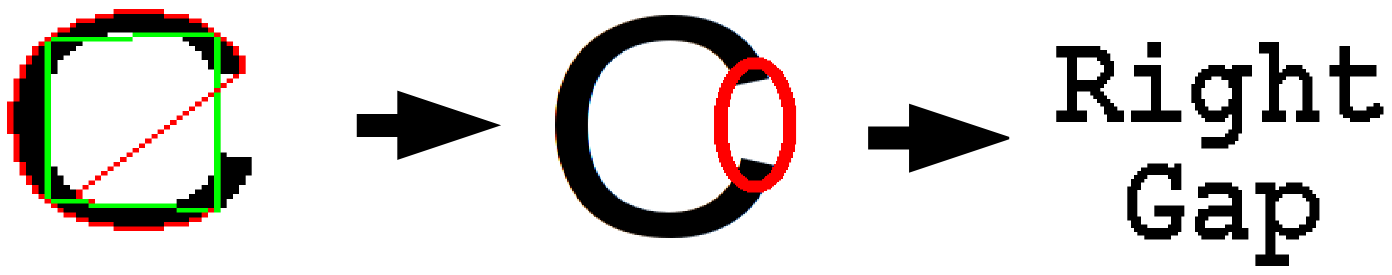
The input for the current prototype is a PNG picture file that contains text in the Courier font. Using the Java BufferedImage class, locations and colors of the pixels in the image can be determined. The program uses these to find the positions of horizontal straight lines of whitespace in the image. It pairs together lines of whitespace and ignores those so that only lines of text remain. Each line is parsed into words using a similar method involving vertical lines. After making spacing analysis, each word is parsed into letters, as seen below.



Then, each letter is converted into a form called a “CharacterModel.” Each model is a collection of “attributes,” currently consisting of a “SectorVector” and a “GapVector.” A SectorVector is formed by parsing the image into portions that pass the vertical line test. After getting rid of unnecessary information, each portion is then transformed into a sum of line segments of different slopes.



A GapVector is simply a collection of the locations of visual “gaps” in the image. Gaps can exist on the four sides (top, right, bottom, and/or left) of the image.



Results

The result is that a letter is simplified into a few pieces of generic information. This procedure is applied to each letter of several different fonts, and information from the results is stored and averaged. Using this, a cache is created to which results from OCR analysis can be dynamically compared.

```
c SectorVector -2 3 GapVector R
```