Spring 4-2015

Accuracy of Optical Character Recognition Software Google Tesseract

Joshua A. Suitter
University of Southern Maine

Accuracy of Optical Character Recognition software Google Tesseract

Author: Joshua Suitter Department of Engineering Advisor: Dr. Mariusz Jankowski, University of Southern Maine

Figure 1. Image of the Constitution

Abstract

Tesseract is an open-source OCR (Optical Character Recognition) software engine originally developed by HP between 1985 and 1995, it is now sponsored by Google Projects (Google Tesseract). While Tesseract is known as one of the most accurate free OCR engines available today, it has numerous limitations that dramatically affect its performance; its ability to correctly recognize characters in a scan or image. During my research I have found that certain fonts are accepted more than others, and font size, spacing, and image quality all play a role in how Tesseract performs. In this project, I will also be looking into Wolfram's Mathematica built-in Tesseract code: Text Recognize. You will see through this project how different fonts, font sizes, image quality, and tilting of an image affect Tesseracts recognition accuracy. The first part of this project I tested the fonts and font sizes using Tesseract. I did error calculations by eye, looking for when a word came back in the text file incorrectly. The reason for using Mathematica's version is so I can automate my error process; getting a more accurate result. In my research, I found that both the original Tesseract program and Mathematica's built-in version are very accurate, especially at higher quality images.

Overview

For this project, I took the first couple sections from the Constitution of the United States of America. I incorporated Microsoft Word to modify fonts and sizes of those fonts to see what affect it had on the documents recognition thru Tesseract. Once these different files were all created in PDF format they were then converted to an image using another free online software. Tesseract takes image files (i.e. .tiff, .jpg,) and extracts the words; as accurately as possible, from the images. Tesseract runs from the command prompt program. I also used Mathematica's Text Recognize code to automate my data and get a more accurate result of how well the OCR works. With other built in codes like Smith Waterman Similarity and Sequence Alignment, I can see how accurately the program is running as well as where the errors occur.

Results

Figure 2 shows the results of researching three different fonts from size 8 to size 16. I chose Times New Roman, Courier New, and Arial for my fonts. One of the biggest results was that the accuracy of Tesseract did depend on the size of the font, as well as the quality of the image created. The font sizes that did the worst were sizes 8 to 10, and all of the "average" quality images (75 dpi or pixels) performed really bad no matter what font or font size. Next I looked into image manipulation and the affect it had on the recognition accuracy. For image tilt, I found that the accuracy of the program really declined above 2 degree of tilt. Blurring an image also declines accuracy, while sharpening increases accuracy to a point.

Font Size	Arial Average	Arial Good	Arial Excellent	TNR Average	TNR Good	TNR Excellent	CN Average	CN Good	CN Excellent	Scanned Tiff	Scanned Jpeg
8	0.00%	59.43%	78.98%	0.20%	9.48%	98.24%	0.00%	20.33%	97.46%	94.04%	94.04%
9	0.00%	95.01%	94.72%	0.29%	87.59%	98.05%	0.00%	56.31%	97.26%	96.09%	96.09%
10	0.29%	95.20%	95.70%	0.59%	97.46%	97.85%	0.00%	74.00%	97.07%	96.97%	96.97%
11	0.59%	95.21%	97.07%	0.39%	96.68%	97.46%	0.49%	87.29%	96.48%	97.17%	97.17%
12	1.27%	96.19%	97.07%	0.59%	96.87%	97.65%	0.49%	96.29%	95.89%	96.38%	96.38%
14	22.68%	95.50%	96.29%	0.59%	96.68%	96.87%	3.13%	72.14%	80.55%	96.29%	96.29%
16	12.81%	94.92%	95.50%	30.21%	96.77%	96.87%	57.58%	70.19%	94.53%	96.48%	96.48%
									Size 18	96.09%	96.09%
									Size 20	95.70%	95.70%

Figure 2. Table of Data from using Mathematica, Text Recognize, and the Smith Waterman.

Conclusions

- Tesseract is a very accurate OCR based program which responds quickly to images with hundred of words.
- There is so much training that could be done to improve this software, and with the right tools it's trainable.
- Tesseract is really affected by the quality of the image that is sent into the program, the worst qualities have the least accurate performance.
- Font sizes also make a difference with the smaller and larger fonts being less accurate. The more common font sizes (10-14) make the OCR perform at its best.
- Image manipulation also can affect the OCR's accuracy and you must be careful when scanning in a document and trying to use Tesseract to extract the words.

Acknowledgements
I would like to thank Dr. Jankowski for all of his contributions to this project.