An Overview of the Tesseract OCR Engine

Ray Smith, Google Inc.

Presented By,
Azmain Adel (1405075)
Ajoy Das (1405079)

May 12, 2017

Department of Computer Science & Engineering,
Bangladesh University of Engineering & Technology

Outline

1. Introduction
2. Architecture
3. Line and Word Finding
4. Word Recognition
5. Static Character Classifier
6. Linguistic Analysis
7. Adaptive Classifier
8. Result
9. Conclusion

Introduction

Introduction

- What is an OCR Engine?

Introduction

- What is an OCR Engine?
- What is Tesseract and what does it do?

Introduction

For almost two decades, **Optical Character Recognition** or OCR systems have been widely used to provide automated text entry into computerised systems.

- Conventional OCR systems never overcame their inability to read more than a handful of type fonts and page formats.
  They were unable to capture-
  1. Proportionally spaced type.
  2. Laser printer fonts.
  3. Many non-proportional typewriter fonts.
- They never achieved major impact on the total number of documents needing conversion into digital form.

Introduction

Early Lessons:

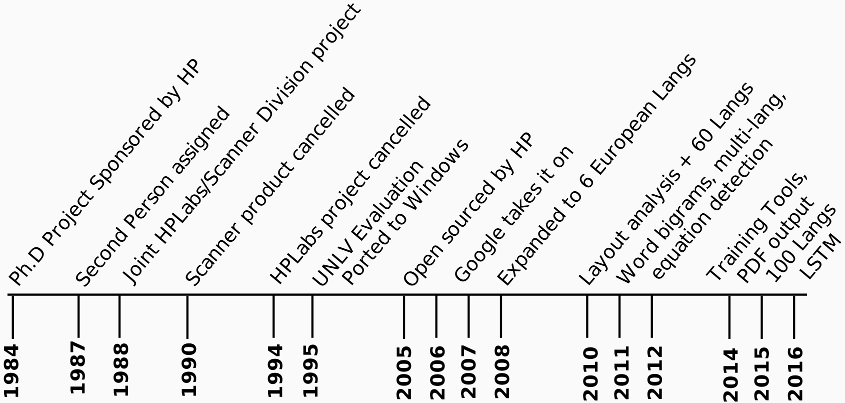
- Be able to OCR your own sales literature!
- OCR should distinguish text from non-text.
- OCR should read white-on gray and black-on-gray as easily as black-on-white.

Decision:

Extract outlines from grayscale images and features from the outlines.

- Profound impact.
- Key differentiator.

History



Ph.D Project Sponsored by HP

Second Person assigned

Joint HPLabs/Scanner Division project

Scanner product cancelled

HPLabs project cancelled

UNLV Evaluation
Ported to Windows

Open sourced by HP

Google takes it on

Expanded to 6 European Langs

Layout analysis + 60 Langs

Word bigrams, multi-lang,
equation detection

Training Tools,
PDF output
100 Langs
LSTM

1984
1987
1988
1990
1994
1995
2005
2006
2007
2008
2010
2011
2012
2014
2015
2016

Figure 1: History of the Tesseract OCR engine.

Architecture

Architecture

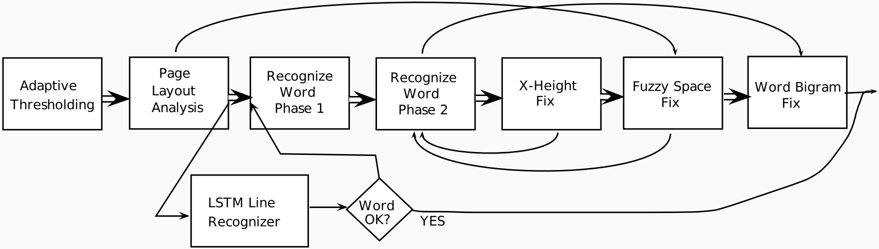
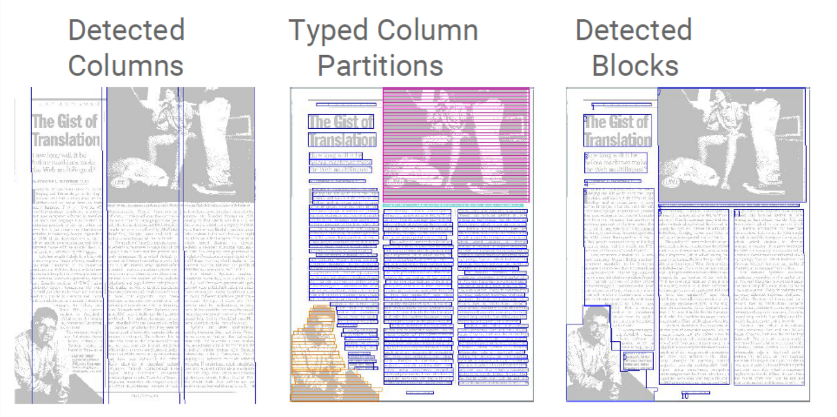


Figure 2: Architecture of the engine.

Page Layout Analysis



Detected Columns

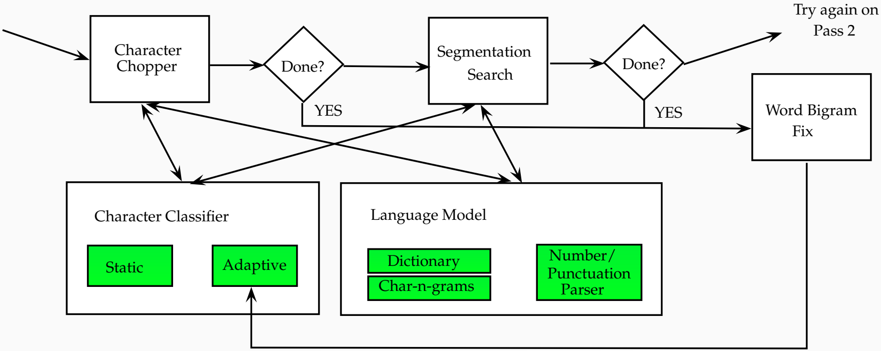
Typed Column Partitions

Detected Blocks

The Gist of Translation
How long will it be before machines make the Web multilingual?

Detecting the layout of a page.

Tesseract Word Recognizer



Recognizing a word in Tesseract.

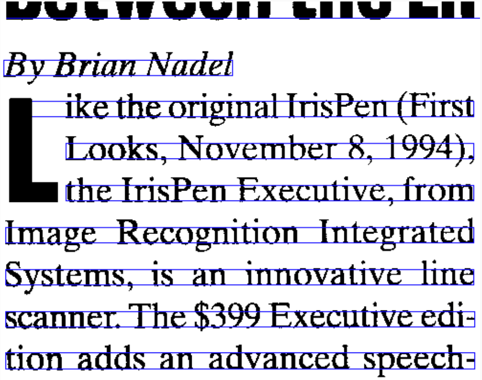
Line and Word Finding

Line Finding

Three main problems in finding a line.

1. Drop caps.
2. Skew lines.
3. Touching lines.

1. Drop Caps

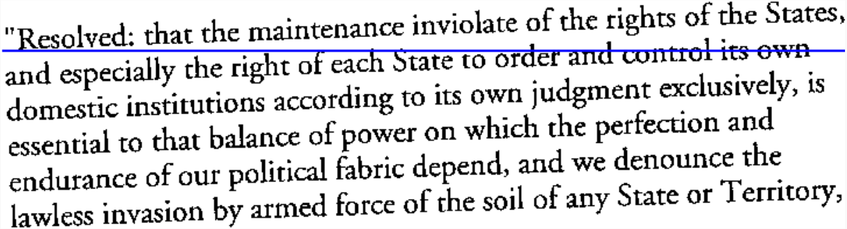


By Brian Nadel

Like the original IrisPen (First Looks, November 8, 1994), the IrisPen Executive, from Image Recognition Integrated Systems, is an innovative line scanner. The $399 Executive edition adds an advanced speech-

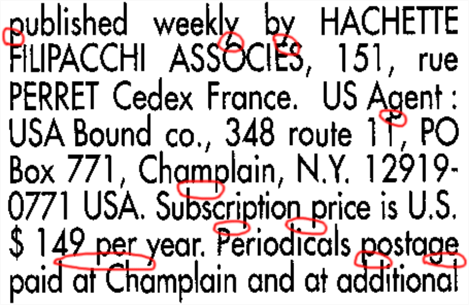
Here, the capitalized letter **‘L’** is preventing from detecting the lines.

2. Skew Lines



"Resolved: that the maintenance inviolate of the rights of the States,
and especially the right of each State to order and control its own
domestic institutions according to its own judgment exclusively, is
essential to that balance of power on which the perfection and
endurance of our political fabric depend, and we denounce the
lawless invasion by armed force of the soil of any State or Territory,

3. Touching Lines



published weekly by HACHETTE
FILIPACCHI ASSOCIÉS, 151, rue
PERRET Cedex France. US Agent :
USA Bound co., 348 route 11, PO
Box 771, Champlain, N.Y. 12919-
0771 USA. Subscription price is U.S.
$ 149 per year. Periodicals postage
paid at Champlain and at additional

Lines touching each other on the points marked as red.

Solution for Line Finding

The OCR assumes that page layout analysis has already provided text regions of a roughly uniform text size.

- A simple percentile height filter removes drop-caps and vertically touching characters.
- The median height approximates the text size in the region, so it is safe to filter out blobs that are smaller than some fraction of the median height, being most likely punctuation, diacritical marks and noise.

Baseline Fitting

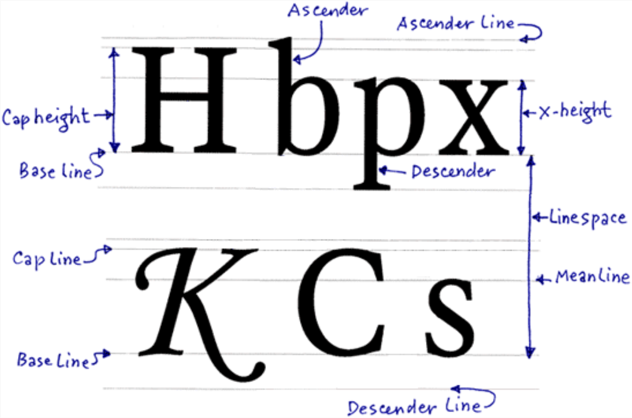
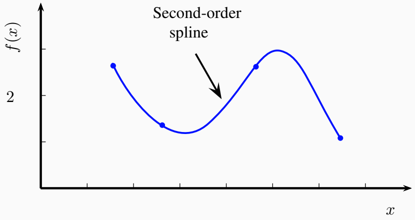


Figure 7: Anatomy of a text line.

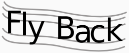
Baseline Fitting

After finding the text lines, the baselines are fitted more precisely using a quadratic spline. This was another first for an OCR system, and enabled Tesseract to handle pages with curved baselines which are common artifacts in scanning and at book bindings.

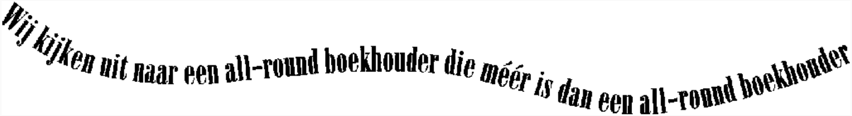


Baseline Fitting

The quadratic spline has the advantage that this calculation is reasonably stable.

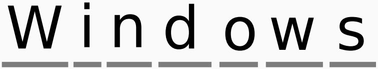


But the disadvantage is that, discontinuities arise when multiple spline segments are required. Tesseract **can not detect** lines like -



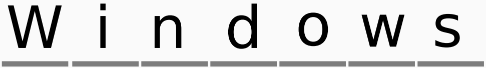
Wij kijken uit naar een all-round boekhouder die méér is dan een all-round boekhouder

Pitch Detection and Chopping



Windows

Proportional Pitch

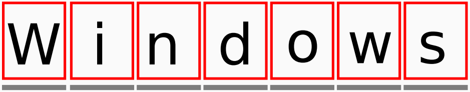


Windows

Fixed Pitch (Monospacing)

Pitch Detection and Chopping

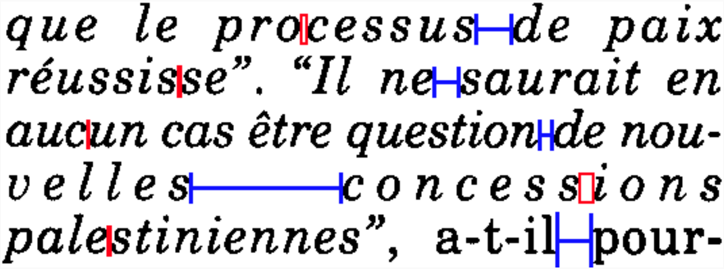
Tesseract finds fixed pitch text using the text lines.
It chops the words into characters using the pitch, and disables the chopper and associator on these words for the word recognition.



Fixed Pitch (Monospacing)

Word Recognition

Word Recognition



que le processus de paix réussisse”. “Il ne saurait en aucun cas être question de nouvelles concessions palestiniennes”, a-t-il pour-

- Tesseract solves most of these problems by measuring gaps in a limited vertical range between the baseline and mean line.
- Spaces that are close to the threshold at this stage are made fuzzy.

Chopping Joined Characters

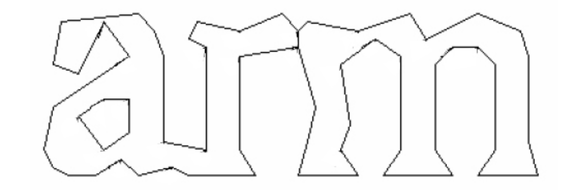


Figure 8: The characters **a, r** and **m** are joined together.

Chopping Joined Characters

Candidate chop points are found from concave vertices of a polygonal approximation of the outline, and may have either another concave vertex opposite, or a line segment.

Chops are executed in priority order. Any chop that fails to improve the confidence of the result is undone, but not completely discarded so that the chop can be re-used later by the associator if needed.

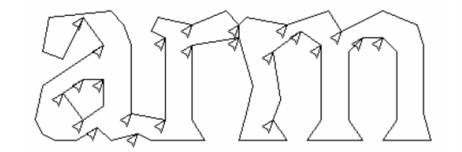
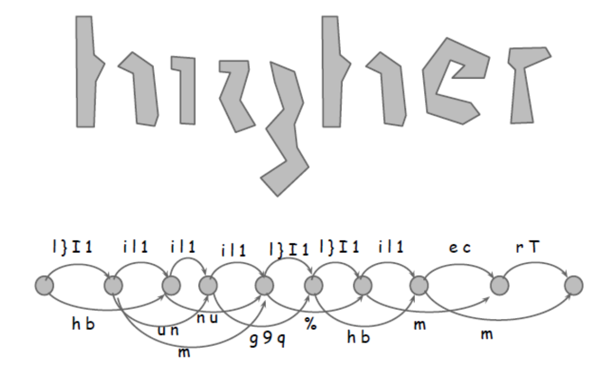


Figure 9: Chopped characters.

Associating Broken Characters

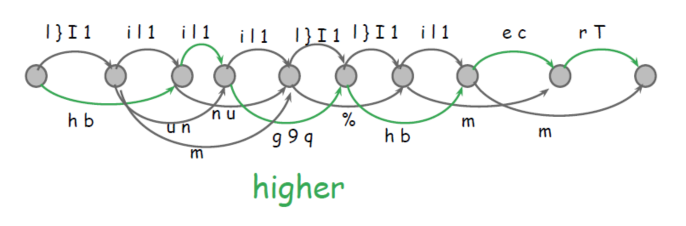
When the potential chops have been exhausted, if the word is still not good enough, it is given to the associator.

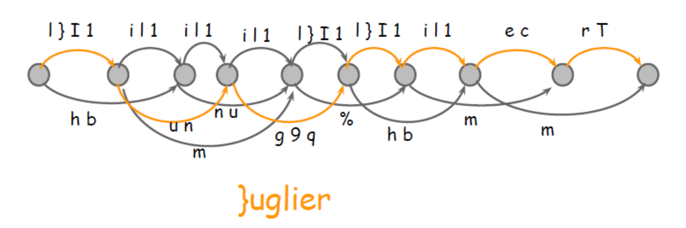


Associating Broken Characters

The associator makes an A* (best first) search of the segmentation graph of possible combinations of the maximally chopped blobs into candidate characters. It does this without actually building the segmentation graph, but instead maintains a hash table of visited states.

Associating Broken Characters





Static Character Classifier

Character Classifiers

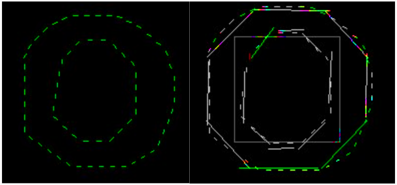
What are Character Classifiers?

- In simple words, a Character Classifier identifies scanned characters and decides what it is.

Character Classifiers

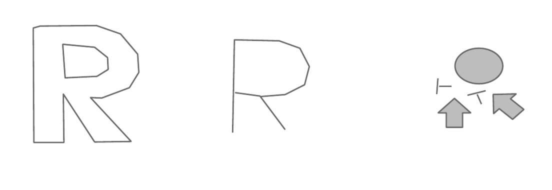
What are Character Classifiers?

- In simple words, a Character Classifier identifies scanned characters and decides what it is.
- It detects **features**, **classifies** it and matches it with **training data**.



First Version

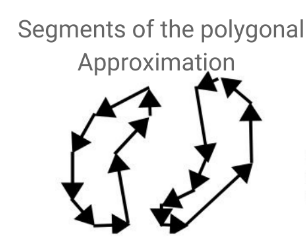
- Based on Topological features.
- Independent of fonts and sizes.



Extracting into topological features.

Second Version

- Based on Polygonal approximations.
- Not robust for broken or damaged characters.



Detecting a broken ‘O’

Third Version

This is called the breakthrough solution!

- Based on matches between prototype features and unknown features.
- The unit is called **Distance**.
- Can cope with damaged characters.
- Computational cost is very high.

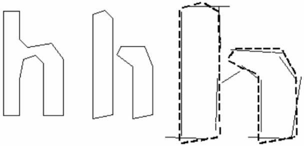
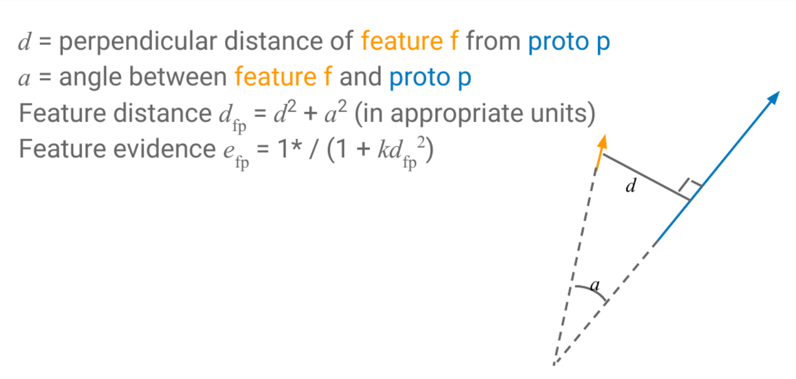


Fig. 6. (a) Pristine ‘h, (b) broken ‘h’, (c) features matched to prototypes.

Measuring the Distance



d = perpendicular distance of feature f from proto p

a = angle between feature f and proto p

Feature distance $d_{\mathrm{fp}} = d^2 + a^2$ (in appropriate units)

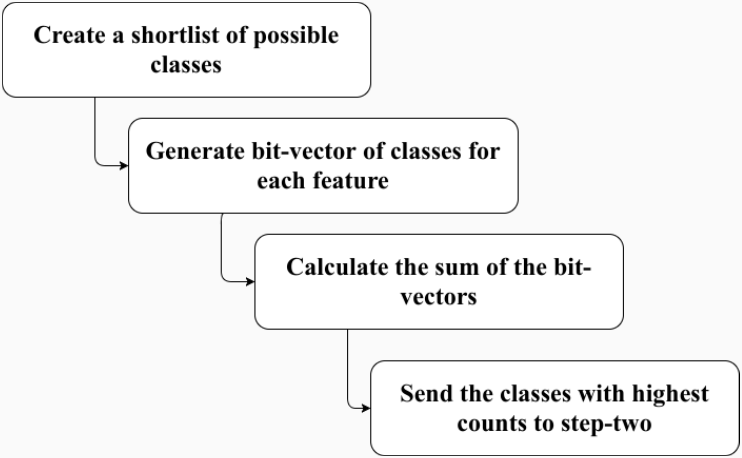
Feature evidence $e_{\mathrm{fp}} = 1^* / (1 + kd_{\mathrm{fp}}{}^2)$

Classification

Classifying characters means to select the character of a already scanned and detected input character.

It is a two-step process.

First Step



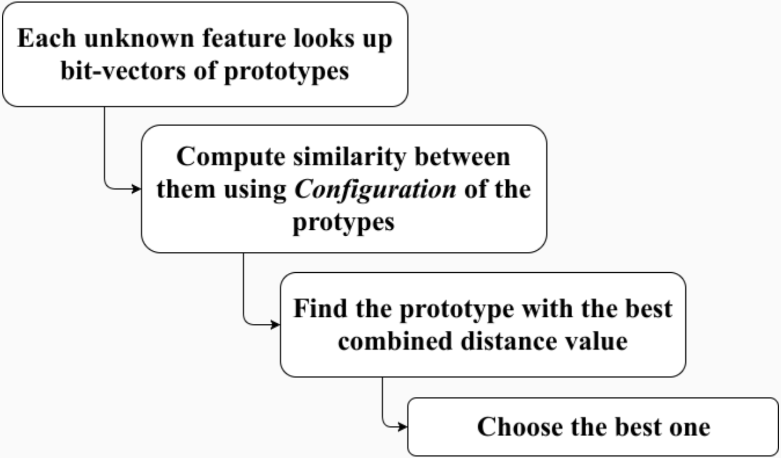
Create a shortlist of possible classes

Generate bit-vector of classes for each feature

Calculate the sum of the bit-vectors

Send the classes with highest counts to step-two

Second Step



Each unknown feature looks up bit-vectors of prototypes

Compute similarity between them using *Configuration* of the protypes

Find the prototype with the best combined distance value

Choose the best one

Training Data

- No need to provide damaged or broken character data.
- The following were provided as input,

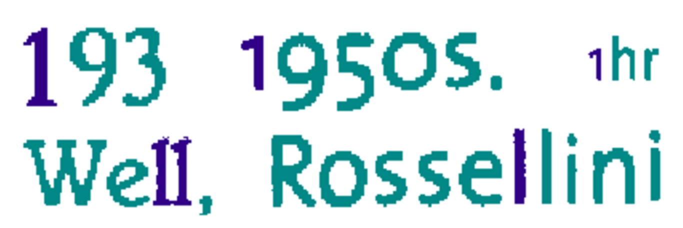
Fonts	8
Attributes	4
Characters	94
Samples	20

- Total of **60160** samples.

Linguistic Analysis

What is Linguistic Analysis?

The linguistic support makes the OCR software faster and more reliable. Tesseract contains little lingiustic analysis.



Linguistic analysis helps in differentiating between **'1'** and **'l'**

[1]

Analysis

The linguistic module chooses the word string from the following categories:

- Top frequent word.
- Top dictionary word.
- Top numeric word.
- Top UPPER case word.
- Top lower case word.
- Top classifier choice word.

The word with the lowest distance rating is chosen.

Problems in Linguistic Analysis

Sometimes it is hard to compare two words from different segmentaions. Tesseract uses two numbers to solve this problem.

1. **Confidence**

$$confidence = foundDistance - normalDistanceFromPrototype$$

2. **Rating**

$$rating = normalDistanceFromPrototype\ x\ totalOutlineLengthofUnknown$$

Adaptive Classifier

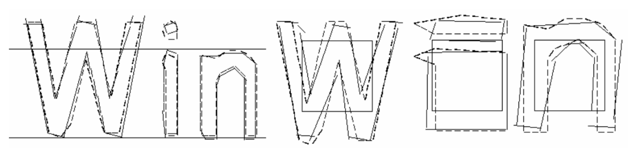
Adaptive Classifier

Static Classifier can benefit from using a Adaptive Classifier.

- Static Classifier is weak in discriminating between characters.
- Font-sensitive Adaptive Classifiers are used to obtain greater discrimination.

Adaptive Classifier

- Static Classifier normalizes characters by moments of size and position.
- Adaptive Classifier uses isotropic baseline normalization.



Baseline and Moment nomalized by using Static and Adaptive classifers.

Baseline normalization helps to distinguish between upper-case and lower-case characters, as well as sub-scripts and super-scripts.

Result

Result

		Character			Word		
Ver	Set	Errs	%Errs	%Chg	Errs	%Errs	%Chg
HP	bus	5959	1.86		1292	4.27	
2.0	bus	6449	2.02	8.22	1295	4.28	0.15
HP	doe	36348	2.48		7042	5.13	
2.0	doe	29921	2.04	-17.68	6791	4.95	-3.56
HP	mag	15043	2.26		3379	5.01	
2.0	mag	14814	2.22	-1.52	3133	4.64	-7.28
HP	news	6432	1.31		1502	3.06	
2.0	news	7935	1.61	23.36	1284	2.62	-14.51
2.0	**Total**	59119		-7.31	12503		-5.39

Table 1: Results of Current and Old Tesseract.

[2]

Conclusion

Summary

The *Tesseract OCR Engine* is now behind most of the commercial OCR engines. But it is still considered as a pioneer in Optical Character Recognition systems.
You can find it here,

`https://github.com/tesseract-ocr/tesseract`

References I

- www.how-ocr-works.com.

- S. Rice, F. Jenkins, and T. Nartker.
  The Fourth Annual Test of OCR Accuracy.
  Technical Report 95-03, pages 403–422, July 1995.

Thank you.

Any Questions?