SALTcymru

Document 5

An overview of the Tesseract OCR (optical character recognition) engine, and its possible enhancement for use in Wales in a pre-competitive research stage

Prepared by the
Language Technologies Unit (Canolfan Bedwyr), Bangor University

April 2008

PRIFYSGOL BANGOR UNIVERSITY
1884

Llywodraeth Cynulliad Cymru
Welsh Assembly Government

This document was prepared as part of the SALT Cymru project, funded by the Welsh Assembly Government under the Knowledge Exploitation Fund's Knowledge Exchange Programme, reference HE 06 KEP 1002

What is OCR technology?

OCR technology allows the conversion of scanned images of printed text or symbols (such as a page from a book) into text or information that can be understood or edited using a computer program. The most familiar example is the ability to scan a paper document into a computer where it can then be edited in popular word processors such as Microsoft Word. However, there are many other uses for OCR technology, including as a component of larger systems which require recognition capability, such as the number plate recognition systems, or as tools involved in creating resources for SALT development from print based texts.

Availability

General Availability

Commercial OCR technologies, of which OCR engines is the core component, are widely available. These commercial engines are highly developed and offer considerable accuracy when working with texts from major languages. With English text for example, the top commercial engines have an accuracy of over 98%. Some companies specializing in OCR technologies offer software developer kits (SDKs) which allow software developers to license the use of the OCR technology in their own systems.

Language Availability

As previously mentioned, the accuracy of major-language commercial OCR is very high. This accuracy is achieved through the combination of language independent algorithms for identifying the likely value of a character with language specific information such as wordlists that improve the results of these algorithms.

Commercial OCR technologies rarely include language specific information for less spoken languages such as Welsh. By attempting to identify individual characters these OCR technologies will still work to some degree with these languages, but the lack of language specific information to compare these results to leads to a considerable drop in accuracy. Unfamiliar or undefined characters such as Welsh's *ŵ* and *ŷ* may not be correctly recognized at all. In lengthy texts, the post-editing required after using OCR technologies may be tedious and time-consuming.

Price

Commercial OCR software is expensive. Market leader Nuance's Omnipage 16 currently retails for £80 per copy for the standard version, and £292 per copy for the Pro version. The company also offers licences to use their OCR engine in the form of an Omnipage SDK, the price of which can be prohibitive for SMEs involved in smaller projects.

Use

Home and Office

OCR technology is commonly used in home and office environments, where the ability to convert printed paper documents into editable electronic documents is a considerable time saver when the only alternative is to redraw or retype a document in its entirety.

Accessibility
The digitization of printed documents can be of enormous benefit to visually impaired users by enabling printed texts to be digitized and read out loud using text-to-speech technologies.

OCR Components in larger systems
OCR engines are often found as components of larger systems that are designed to track information using visual cues that have been placed on objects. An example of this is the technology used to identify the number plates of cars entering and leaving congestion zones. Similarly, OCR technology can also be used track the progress of a delivery or the progress of a component through a supply chain.

Creating Corpora and Lexica
OCR technology is also invaluable to developers that are involved in the creation of resources used by speech and language technologies. By digitizing print-based texts, developers can create electronic resources such as corpora and lexica for languages where existing digital texts are insufficient, unsuitable or do not exist. It is from corpora and lexica that resources such as word lists and grammar rules are generated. These resources lie at the heart of SALT development for any particular language.

Creating Translation Memory from printed texts
OCR technology can be of great benefit to translators as they move over to using Computer Assisted Translation (CAT), as it allows the creation of valuable translation memories from previous translations which were archived in paper form. The use of such translation memories can increase translator productivity by up to 40%.

Why is OCR of interest to SALT Cymru?

Pre-competitive advantage
The development and refinement of open source OCR technology would enable developers to flexibly and cheaply incorporate OCR technology into their systems without the burden of developing or licensing the underlying technology. By lessening the overheads involved in the development of such systems, smaller sized enterprises such as SMEs could consider moving into markets where previously only larger companies were able to compete, especially if the relevant training was also made available.

Language Support for Welsh
As mentioned previously, highly-developed OCR engines tend to only be available for major languages. This means that most of the world's languages are currently not well supported, providing an opportunity for companies wishing to specialize in providing support for these unsupported languages.

There is, for instance, currently no OCR technology in existence that produces satisfactory results when scanning Welsh and bilingual Welsh/English printed text. This is a major problem for those wishing to digitize printed texts that contain Welsh or a combination of Welsh and English (see the PowerPoint presentation given by

representatives of the National Library of Wales at the JISC Digitization Conference, 2007. Link: www.jisc.ac.uk/media/documents/programmes/digitisation/jiscdigicon07locock.ppt). The ability to accurately digitize Welsh language texts would be of great benefit to many sectors and would enable:

- The ability to create Welsh digital language resources such as lexica and corpora for Speech and Language Technologies developers from printed resources
- Easier digitization of historical Welsh texts as undertaken by the National Library of Wales (cf. projects such as Culturenet's Books from the Past)
- The ability to process forms returned in Welsh, such as those from the Welsh Assembly Government and other public bodies
- The ability to enable blind users of Welsh text-to-speech to have access to books not available in digital form
- The creation of Welsh/English Translation Memories from existing parallel translations that survive only in printed form

Expertise developed in the process of developing OCR tools for the Welsh language could be put to commercial use with other languages that lack full OCR support. A long tail of the world's languages are in a similar position to that of Welsh.

Tesseract OCR Engine

What is Tesseract?

Tesseract is an open source optical character recognition (OCR) engine originally developed at Hewlett-Packard between 1985 and 1995, but never commercially exploited. It rated highly at The Fourth Annual Test of OCR Accuracy (http://www.isri.unlv.edu/downloads/AT-1995.pdf) held in 1995 at the University of Nevada, Las Vegas' Information Science Research Institute (ISRI: http://www.isri.unlv.edu/). However by that time, Tesseract's development had ceased.

In 2005, HP transferred Tesseract's unaltered code to the ISRI and it was released as open source. ISRI discovered that the original developer, Ray Smith (see http://research.google.com/pubs/author4479.html), was now employed at Google after several years working on the market leading commercial OCR engine *Omnipage*. Google were persuaded by ISRI to allow Smith to continue development of Tesseract as open source software. Version 2.0 is now available for download from Google Code at http://code.google.com/p/tesseract-ocr/.

Limitations of Tesseract

Tesseract is an OCR engine, not a complete OCR program

Tesseract is an OCR engine rather than a fully featured program similar to commercial OCR software such as Nuance's Omnipage. It was originally intended to serve as a component part of other programs or systems. Although Tesseract works from the

command line, to be usable by the average user the engine must be integrated into other programs or interfaces, such as FreeOCR.net, WeOCR or OCRpous. Without integration into programs such as these, Tesseract has no page layout analysis, no output formatting and no graphical user interface (GUI).

OCRopus
OCRopus is an open source document analysis and OCR system also funded by Google. It provides much of the layout analysis functionality missing from Tesseract. It is also able to use engines other than Tesseract. See: http://code.google.com/p/ocropus/.

WeOCR
WeOCR is a platform for Web-enabled OCR, which provides users with an online interface for OCR engines, including Tesseract, which allows users to upload images of English text in bmp, jpeg and pbm/pgm/ppm formats and receive the output in a text file format. It can be accessed from the following link:
http://asv.aso.ecei.tohoku.ac.jp/tesseract/ .

FreeOCR.net
FreeOCR.net is a simple but effective freeware program that uses Tesseract as its OCR engine and produces accurate results from print, via your scanner to text format when scanning English texts. It does however lack layout analysis and output formatting, and although available as freeware, FreeOCR.net is not open source. Nevertheless, it serves as an impressive and foolproof demonstration of the potential of the Tesseract engine. It can be downloaded from: http://softi.co.uk/freeocr.htm.

Unsupported features
Although Tesseract has been modified to deal with UTF-8 characters, Tesseract may not work well with languages that possess complex characters, or connected scripts such as Arabic. Only left-to-right scripts are supported. Right-to-left texts are currently processed as if they were as if they were left-to-right texts. ASCII punctuation and digits are expected by the code, so any language using alternatives to these will not be fully supported.

How does Tesseract work?

A comprehensive overview of the Tesseract OCR Engine entitled *An Overview of the Tesseract OCR Engine* by Ray Smith is available from the IEEE, at the following address:

http://ieeexplore.ieee.org/iel5/4376968/4376969/04376991.pdf?tp=&isnumber=4376969&arnumber=4376991 (Subscription or payment may be required)

For convenience, the following is a brief overview of how Tesseract works:

1. Outlines are analysed and stored
2. Outlines are gathered together as *Blobs*
3. Blobs are organized into text lines
4. Text lines are broken into words
5. First pass of recognition process attempts to recognize each word in turn
6. Satisfactory words passed to adaptive trainer

7. Lessons learned by adaptive trainer employed in a second pass, which attempts recognize the words that were not recognized satisfactorily in the first pass
8. Fuzzy spaces resolved and text checked for small caps
9. Digital texts are outputted

During these processes, Tesseract uses:

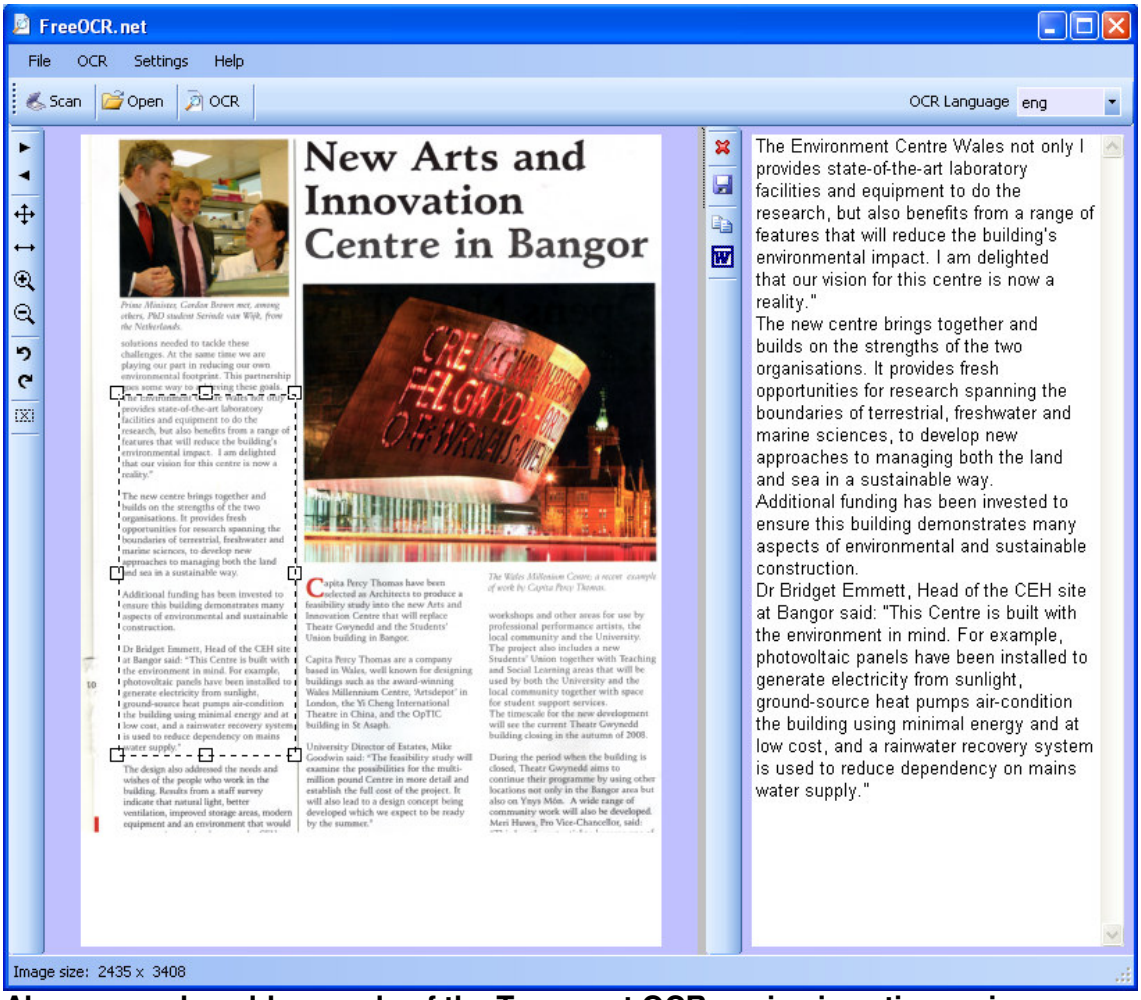
- algorithms for detecting text lines from a skewed page
- algorithms for detecting proportional and non proportional words (a proportional word is a word where all the letters are the same width)
- algorithms for chopping joined characters and for associating broken characters
- linguistic analysis to identify the most likely word formed by a cluster of characters
- two character classifiers: a static classifier, and an adaptive classifier which employs training data, and which is better at distinguishing between upper and lower case letters

The project page for Tesseract is located on Google code at the following address: http://code.google.com/p/tesseract-ocr/

This page features background, details of important changes and supported platforms in addition to a project roadmap and a list of developers. Downloads of the Tesseract engine, as well as associated files and utilities are also located her, and an associated Google Group can be found at http://groups.google.com/group/tesseract-ocr.

How accurate is Tesseract OCR?

The above processes ensure that Tesseract is highly accurate when recognizing texts from languages that are currently supported. Results from The Fourth Annual Test of OCR Accuracy (http://www.isri.unlv.edu/downloads/AT-1995.pdf) are still available online, where, for example, Tesseract demonstrated a Word Accuracy of 97.69% with a sample of English newspapers. Since these tests, the Tesseract development team at Google claim to have improved Tesseract's general results by 7.31% for Tesseract version 2.0.



Above: a real world example of the Tesseract OCR engine in action, using FreeOCR.net

What are the language-specific components of Tesseract?

For a language such as English, 8 components are used:

1. General Words Wordlist (`tessdata/eng.word-dawg`)
2. Frequent Word Wordlist (`tessdata/eng.freq-dawg`)
3. User Wordlist (`tessdata/eng.user-words`)
4. Index for Character Set (`tessdata/eng.inttemp`)

5. Box file – for use in locating characters in the training file (tessdata/eng.normproto)
6. Box file – for use in locating characters in the training file (tessdata/eng.pffmtable)
7. Language's Character Set (tessdata/eng.unicharset)
8. Character Cluster Disambiguator - for 'm' and 'rn', for instance. (tessdata/eng.DangAmbigs)

OCR technology uses character recognition to attempt to identify the individual characters that make up a printed text. Although the process used to identify individual characters is language independent, Tesseract must be given a list of the specific characters used by a language (item 4 in the list above).

Tesseract must then be trained to correctly identify these characters when they appear within a piece of text. Training is done by feeding into Tesseract a document with words, sentences, symbols and numbers from the required language which contains a recommend ten to twenty example of each of the characters used by that language. Such a list has been added to this document as an appendix. This list must be fed in twice, once as digital text and once as a scan of a printed version of the same text. This produces a 'boxfile' containing Tesseract's interpretation of the position of characters and their identity.

The next part of the process is to manually correct any errors made by Tesseract, for example the identification of ŵ as W or the identification of the letter combination *rn* as *m*. A useful utility with a graphical user interface now exists to simplify this process, and is available from the Tesseract project page. Once this task has been finished, common mistakes such as those mentioned above can be added to the Character Cluster Disambiguator file. This training process must be repeated with all font types required, including bold, italic and underlined versions of the same font. The Character Cluster Disambiguator file, in conjunction with a language's word list, helps Tesseract identify a word by suggesting possible corrections to certain characters that allow Tesseract to locate the correct word in its word list. For example, the file can be used to suggest to Tesseract that *rn*, *wr*, *iii*, and *an* could all potentially be misidentifications of the letter *m*, and Tesseract will search the wordlist accordingly.

However, not all languages will have a list of the commonly used words at their disposal. A list of the head words from a dictionary, for example, is not sufficient as all inflected forms must also be included. For example, *mouse* and *mice* should both be included in an English wordlist, and so too *run* and *ran*. Many other languages undergo far more inflection than English, so their corresponding wordlists are likely to be both longer and harder to create. In Welsh for example, nouns like *coffi* (coffee) occur regularly as *goffi*, *choffi* and *choffi,* effectively quadrupling the number of nouns in a list. Many European languages have significantly more verbal forms compared with English. This inherent complexity in language is part of the reason that resources such as wordlists have not been develop for many languages with less resources. Bespoke wordlists would have to be created for any language supported where wordlists are not available. In truth, for optimum performance, Tesseract requires not one, but two word lists. One should contain the most frequently used words in a language, which Tesseract will search first, the second, which Tesseract will only search after failing to find a word in the first list, should contain the less frequently used words in a language. A third list for user-added words also exists.

In theory, the above steps should allow for the creation of an OCR engine in languages currently unsupported by Tesseract. However, some languages may not be suitable candidates, as right to left languages are currently not compatible with some of

the hardcoded functionality built into Tesseract. Depending on character sets, some languages with complicated glyphs or characters may also be unsuitable. However, Google are currently working on increased language support in future versions of Tesseract.

Development

Tesseract development is currently being led by Google, under the direction of Ray Smith, Tesseract's original developer and one of the foremost experts on OCR technology. Although the participation of Google is of obvious benefit to the project, detailed information concerning the exact nature of the work being undertaken by Google's software engineers between updates is not made publicly available. It is therefore difficult for independent developers to coordinate their work with that being done by Google, and presently developers cannot be certain that their work will be compatible with, and not duplicate, work already carried out by Google.

An example relevant to language development is the following statement made by the lead developer, Ray Smith, at Google:

> "Although it is very tempting to try to expand tesseract to new languages, if you did so, you would be overlapping significantly with the work going on at Google. Of course that leaves anyone that wants a different language in the difficult position of either waiting for it to be available, or trying to train it themselves. I will be in a much better position to discuss language compatibility after the next release, by which time there will be much more language support."

It would therefore seem a sensible precaution for anyone intending to develop the language aspect of Tesseract to first attempt to liaise with the developers at Google.

Tesseract Roadmap
(This roadmap is taken directly from Tesseract's page on Google Code. Accessed 29/03/08.)

Version 2.00 is now available and contains the following new features:

- Support for English, French, Italian, German, Spanish, Dutch
- Scripts to test accuracy against the original 1995 tests run by UNLV
- Ability to train in other languages and scripts

We are considering the following features for upcoming releases:

- ground truth data release
- integration with OCRopus (http://www.ocropus.org/), to support layout analysis
- integration with Leptonica (http://www.leptonica.com/), to support layout analysis and more image formats
- support for even more languages
- high-resolution character shape modelling for improved recognition rates
- a GUI frontend (again, probably shared with OCRopus)

Licence

Originally developed by Hewlett-Packard, Tesseract was released under the Apache 2.0 open source licence ten years after its development had come to an end. The Apache 2.0 open source licence is considered a free software license, compatible with version 3 of the GPL, by the Free Software Foundation. It allows the freedom to use the software for any purpose, including its distribution, its modification, and the distribution of modified versions of the software. However, unlike LGPL licences, the Apache 2.0 license does not require modified versions of the software to be distributed under the same license as the original software. This allows the direct commercial exploitation of modified versions, making the software attractive to business whilst at the same time not undermining the usefulness of the original software to those wishing to develop open source products.