A Platform for Web-Based OCR Systems with Server Search Function

Hideaki Goto

Information Synergy Center, Tohoku University, Japan

1 Introduction

This report introduces a platform for Web-based OCR (Optical Character Recognition/Reader) systems that allows end users to search for and use OCR servers over networks and is expected to accelerate the developments of OCRs and related applications as well.

Recent OCR software and systems are becoming more and more complicated and the development requires expertise in various fields of researches. For example, those who are studying character recognition need to have some knowledge about language processing as well. Document layout analysis does benefit from the feedback from character recognition stages. It has become very difficult for a researcher or a small group of people to build and study a complete system using OCR technologies.

Many developers working on layout analysis have been using commercial OCR libraries, most of which are expensive, or doing laborious coding of OCR engines , despite they wish to use high-performance OCR engines with cutting-edge technologies. As the applications of OCR have recently been extended widely and seem to be still growing, the problems above are becoming great obstacles in studying and developing various advanced systems such as multilingual OCRs, translation glasses, etc.

Even within one field of research, people often have some problems in evaluating their algorithms. Researchers are often requested to perform comparisons of the new algorithms with some conventional algorithms. The task could be quite laborious because many of the recent sophisticated algorithms are difficult to implement. To make matters worse, some technical papers do not provide important information needed to implement the algorithms.

One of the possible solutions to these problems is to share software components among researchers and developers. Actually, some people have been releasing their programs as Free Software. However, some other problems still exist. Firstly, the number of such programs is very limited. Many researchers think the programs are a kind of intellectual property and are not willing to release the source codes even after the related papers have been published. Secondly, even if the programs are available, they are not always easy to install due to lacking libraries, different operating systems, version mismatch, coding problems, etc. Third, freely available programs in academic fields are not so easy to find over the Internet, since there are few directories specialized for them.

Another possible solution is to use Web Application Servers (WAS). Since the programs run on the web server side, people can provide computing (pattern recognition) power to others without having their source codes or executables being open [1]. A system for distributed object programming and a prototype system for a text locating competition are described in [1]. In 2004, we proposed a framework called "Synergetic OCR" that allows people to use OCR services over networks [2]. Many OCR engines work cooperatively in the framework and the system is expected to yield some important features such as accuracy improvement by the majority logic [2–4], multilingual processing, etc. Although the framework was originally intended to be used by end users, it is expected to be useful for developers and researchers as well.

We developed a platform to make the deployment of Web-based OCRs easy and to enable people to search for OCRs that match their requirements. Section 2 describes the toolkit for Web-based OCR, and Section 3 describes a directory-based server search system.

2 Toolkit for Deployment of Web-Based OCR

Making a Web-based OCR from scratch as an application server requires a lot of expertise about network programming and network security. Since many OCR developers and researchers are not so familiar with network programming, we have developed a toolkit to help those people build secure Web-based OCRs easily. The toolkit consists of some interface programs and filter programs that work together with the web server. Figure 1 shows the block diagram of it.

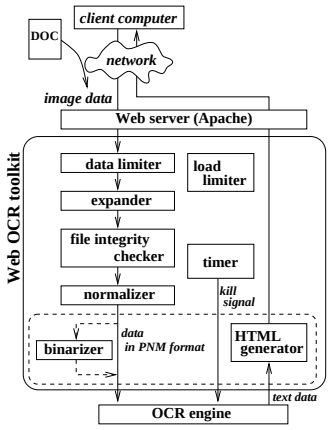

Fig. 1. Block diagram of Web OCR toolkit

The document image is sent from the client computer to the toolkit via a web server. We used the Apache HTTP server. The data limiter limits the size

of the input data to protect the server from being stopped by corrupt data or malicious data such as extremely large ones.

The users must be frustrated if it takes very long until they receive recognition results. We need a mechanism that prevents the server from processing too many images at the same time. The load limiter provides the mechanism. The expander is used to expand the data if it is compressed. This module also has a limiter to avoid explosion of wrong data. The file integrity checker examines the correctness of the image file. Some file header information is examined.

It is not practical to make every OCR engine accept various types of image format. It may be a good idea to absorb the file format differences in the toolkit. Since the PNM format (including PBM, PGM, and PPM) has been widely used and many Open Source OCR software support it, we chose the format as a common file type. When the toolkit receives an image data in Windows and OS/2 Bitmap (BMP) or JPEG, the normalizer converts the data into PNM.

The normalized image is passed to the OCR engine. A small shell script is used to let the OCR developers configure the options of their engines. A binarizer in the toolkit can be used if the OCR engine does not accept color or gray-scale images. Finally, the recognition results are converted into HTML data and sent back to the client.

The timer is a very important module to make the server secure and stable. Many users abort the sessions by pressing the web browser's stop button when it takes long to send data or to have the recognition results. The OCR engine should stop as soon as the session is aborted. Otherwise, the OCR engine could take most of the server's computing power in vain. OCR engines could continue running if they have some critical bugs. The timer tries to terminate the OCR engine if the engine runs longer than the specified time or the session is aborted.

3 Server Search System

We developed a directory-based server search system to enable users to search for OCR engines. We examined some resource discovery mechanisms including those designed for P2P computing and for Grid computing [5]. We concluded a directory-based one was enough for our purposes since it is simpler, easier to control, and the number of OCR servers would not be so great compared with the number of computing resources in Grid or P2P computing environments.

Figure 2 depicts the server search system. Every OCR server has a specification file in XML (eXtensible Markup Language), which is written by the server administrator. XML has been recognized as one of the most powerful language for data exchanges between computers. Another advantage of using XML files is that humans can easily read the contents using XSLT (eXtensible Stylesheet Language Transformations).

The specification file describes the specifications of the server, including the location (URL) of the server, the name of OCR engine, supported languages, supported document types, etc. The portal server has a robot program for collecting the specification files automatically and periodically from the OCR servers. The

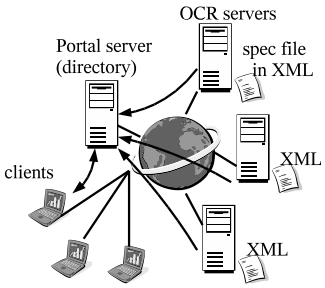

Fig. 2. Directory-based server search system

robot analyzes each specification file in XML and update the database entries. A simple search program picks up the OCR servers that match the client's needs from the database and shows the search results.

4 Conclusions

A simple platform for Web-based OCR systems with server search function has been proposed. A prototype system was implemented and tested under the Linux operating system. The platform will be useful for many OCR developers to share their resources while protecting intellectual properties. We also hope the platform will open up some new applications of character recognition.

We will need a common Web-API (Application Program Interface) for character recognition in order to develop systems using many OCR engines. Our future work includes designing such an API, developing some algorithms that yield synergetic effects of OCR engines, and investigating the potential of Web-based OCRs further.

References

1. Lucas, S.M. and González, C.R.J. : Web-Based Deployment of Text Locating Algorithms. Proc. of First International Workshop on Camera-Based Document Analysis and Recognition (CBDAR2005) 101–107
2. Goto, H. and Kaneko, S. : Synergetic OCR : A Framework for Network-oriented Cooperative OCR Systems. Proc. Image and Vision Computing New Zealand 2004 (IVCNZ 2004) (2004) 35–40
3. Rice, S.V., Kanai, J., and Nartker, T. : A Report on the Accuracy of OCR Devices. ISRI Technical Report, Univ. of Nevada, Las Vegas (1992) 1–6
4. Miyao, H., Nakano, Y., Tani, A., Tabaru H., and Hananoi, T. : Printed Japanese Character Recognition Using Multiple Commercial OCRs. Journal of Advanced Computational Intelligence **8** (2004) 200–207
5. Berman, F., Fox, G.C., and Hey, A.J.G. : Grid Computing — Making the Global Infrastructure a Reality —. Wiley (2003)