IJCSN International Journal of Computer Science and Network, Volume 6, Issue 2, April 2017
ISSN (Online): 2277-5420
www.IJCSN.org
Impact Factor: 1.5

Improving Optical Character Recognition for Low Resolution Images

[1] Mahendra K. Ugale; [2] Madhuri S.Joshi

[1, 2] Department of Computer Science and Engineering,
Jawaharlal Nehru Engineering College, Aurangabad,
M.S. India.

Abstract - Optical Character Recognition (OCR) systems frequently produce errors for images with noise or with low scanning resolution. In this paper, emphasis is given on different OCR technology that can be used to retrieve lower resolution images using different OCR technology. Image restoration using resolution expansion is important in many areas of image processing. Now emerging trend is to have systems to recognize characters in computer system when information is scanned through paper documents which are in printed format related to different subjects. This paper attempts to use Asprise OCR SDK for text extraction for OCR and compares the results with FreeOCR.

Keywords - *Keyword Recognition, Optical Character Recognition, Low Resolution.*

1. Introduction

The process of optical character recognition (OCR) extracts the text in images so that it can be modified and searched. Output of OCR systems often produces errors when the images contain noise or the scanning resolution is low. The optimal resolution to scan images for most OCR systems is 300 dots per inch (dpi). The two OCR engines used are FreeOCR and Asprise which is commercial SDK. **[1]**

Optical character recognition (OCR) of document images continues to be of great importance as there is an attempt to become a paperless society. Restoring text from video surveillance imagery is often crucial to law enforcement agencies**. [2]**

Text image super-resolution is a challenging problem in the computer vision field. In particular, low-resolution images hamper the performance of typical optical character recognition (OCR) systems. **[3]** Several commercial OCR systems with 100% recognition accuracy are available but most of them are suitable for Latin based scripts.

Proposed system is based on implementation of OCR using Asprise SDK.

2. Related Work

Following section shows survey from the other author research papers.

Sanjay Kumar proposed work on offline handwriting recognition, to provide improvement in the ability of recognition accuracy. They use the neural network method to implement the OCR. [1] Paul D. Thouin proposed work on new resolution expansion technique for the restoration of low-resolution grayscale text images.[2] Chao Dong adopted an SRCNN approach in the task of text image super-resolution for facilitating optical character recognition.[3] Imad Qasim Habeeb proposed a method that includes traditional alignment among resulting texts used by related methods; and also no training is needed on errors before executing it [4].

3. Optical Character Recognition

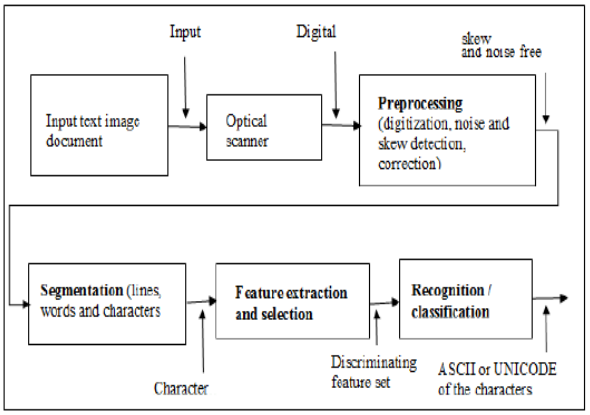

Fig-1 Components of OCR System

The input for the OCR problem is pages of scanned text. To perform the character recognition, the application has to go through three important steps.

1-Optical Scanner

Character images can be acquired from a scanner or any other electronic source. It will be stored in any image format

and that image may be a color, gray scale, but the actual processing will take place on binary images.[6]

2-Preprocessing

The image resulting from the scanning process may contain a certain amount of noise. Depending on the resolution on the scanner and the success of the applied technique for thresholding, the characters may be smeared or broken. Some of these defects, which may later cause poor recognition rates, can be eliminated by using a preprocessor to smooth the digitized characters.

3-Segmentation:

Given input image, identify individual glyphs (basic units representing one or more characters, usually contiguous).

4-Feature Extraction:

From each glyph image, extract features to be used as input of ANN. This is the most critical part of this approach.**[6]**

5-Classification:

The recognition module will recognize the characters by using the features extracted by the feature extraction method. There are several recognition/ classification approaches including template, neural networks, decision tree based, Bayesian based, Support Vector Machines SVM etc.**[6]**

Optical Character Recognition Tools-

This paper is based on testing scanned documents on two trial versions of OCR software.

1. **FreeOCR**
2. **Asprise OCR**

1. FreeOCR

FreeOCR is a free Optical Character Recognition Software for Windows and supports scanning from most Twain scanners and can also open most scanned PDF's and multi page Tiff images as well as popular image file formats. FreeOCR outputs plain text and can export directly to Microsoft Word format.

Free OCR uses the latest Tesseract (v3.01) OCR engine. It includes a Windows installer and it is very simple to use. It supports opening multi-page tiff documents, Adobe PDF and fax documents as well as most image types including compressed Tiff's which the Tesseract engine on its own cannot read . It now can scan using Twain and WIA scanning drivers.

FreeOCR V4 includes Tesseract V3 which increases accuracy and has page layout analysis so that more accurate results can be achieved without using the zone selection tool. **[8]**

Tesseract is chosen as the engine of the OCR because of its widespread approval, extensibility and flexibility, its community of active developers, and the fact that it works out of the box.

2. ***Asprise OCR SDK***

Asprise OCR SDK is a commercial optical character recognition and barcode recognition SDK library that provides an API to recognize text as well as barcodes from images (in formats like JPEG, PNG, TIFF, PDF, etc.) and output in formats like plain text, xml and searchable PDF. **[8]**

The following languages are supported by Asprise version 5 Croatian, Czech, Danish, Dutch, English, Finnish, French, German, Greek, Hungarian, Icelandic, Indonesian, Italian, Malay, Maltese, Norwegian, Polish, Portuguese, Romanian, Russian, Spanish, Swedish or Turkish. MRZ and MICR are supported. **[9]**

Asprise C# .NET OCR (optical character recognition) and barcode recognition SDK offers a high performance API library for you to equip your C# .NET applications (Windows applications, Sliverlight, ASP.NET web service applications, ActiveX controls, etc.) with functionality of extracting text and barcode information from scanned documents.

Features-

- **Highest Level of Accuracy**
  Asprise OCR can easily recognized documents of low resolution.

- **Excellent format retention.**
  Text layouts on input documents are preserved.

- **High Speed**
  ASPRISE OCR uses optimized ocr engine to perform excellent recognition in very short time.

- **Ease of use**
  Complex parameter configurations are removed from Asprise OCR SDK. You only have to supply the image document. Asprise OCR can intelligently determine the best setting internally.
- **Barcode Recognition**
  Beside characters (letters and numbers), Asprise OCR can recognize almost every kind of bar code. You can choose to recognize barcode or characters or both. Currently, the following bar code formats are supported:
  CODE 128 (128b, 128C, 128raw)**.**
  - EAN 8 EAN 13
  - UPC
  - Code3 of 9
  - Codeinterleaved 2 of5.

Table 1- Feature Comparison of tested OCR technology

Cases	Criteria	Nature of Page	Free OCR	Asprise OCR SDK
Case-I	Keyword Recognition Rate	Plain Text Page with 300 dpi resolution.	355/358	356/358
	Accuracy of Recognized Keyword (%)		98%	99%
	Size Before Tagging		1.12 MB	1.12 MB
	Size After Tagging		23 kb (rtf file)	2kb (rtf file)
	Time Required To Recognize 358 Keywords		12 Sec	10 Sec
	Keyword Recognition Rate	Plain Text Page with 150 dpi resolution	150/358	153/358
	Accuracy of Recognized Keyword (%)		41.58%	59.60%
Case-II	Recognition of 2 rows and 3 columns with Keywords	Tabular data document.	All keywords are recognized without layout.	All keywords are recognized with layout.
Case-III	Barcode Recognition rate	Barcode Data Document	NA	Correctly Recognized contents

Table 2-Experimental Results

Criterion	Asprise OCR SDK	Free OCR
Scanner Driver Supported	TWAIN	TWAIN
Table/ Spreadsheet Recognition	✓	✗
Searchable PDF Output	✓	✗
PDF Password Support	✓	✗
Vertical Text Recognition	✓	✓
Barcode Recognition	✓	✗

4. Experimental Results

For testing purpose different cases are considered-

Case 1- Keyword recognition rate for plain text page with 300 dpi resolution and plain text page with 150 dpi resolution.

Case 2- Keyword recognition rate from Tabular data document.

Case 3- Barcode Recognition rate from document.

After performing OCR by FreeOCR and Asprise OCR SDK following Experiment result is found.

Following table shows experimental results of FreeOCR and

Asprise OCR by considering various criteria.

Following figure show GUI for Tagging of documents

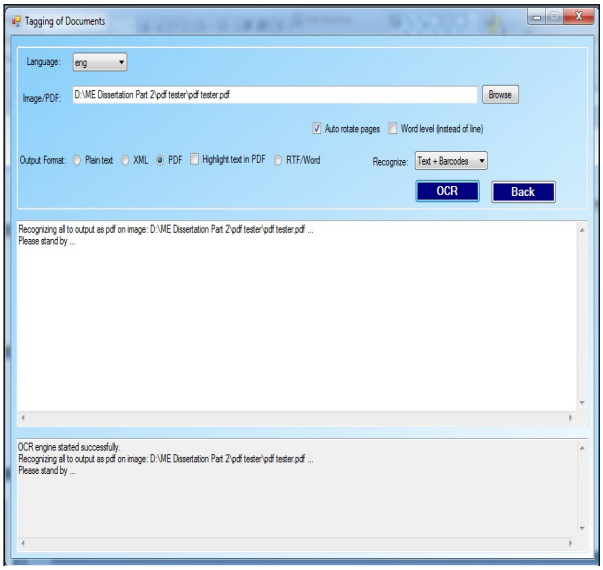

Fig 2 –Tagging of Documents

Algorithm of tagging of documents-

This flowchart explains different steps for tagging of documents.

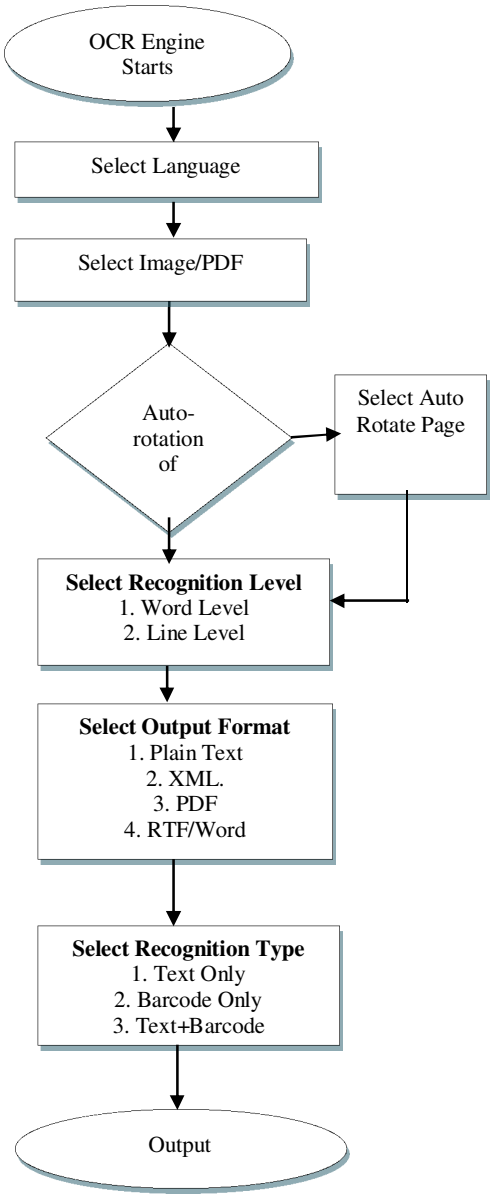

Fig 3 – Algorithm of Tagging of Documents

For effective tagging we have to go through all the phases that are specified in above algorithm.

5. Conclusion

In this paper two different OCR technologies are considered, FreeOCR and implementation based on Asprise OCR SDK. Keyword Recognition Rate from plain text page having 358 Keywords with good Resolution is greater than keyword recognition rate from plain text page having 358 Keywords with low Resolution. Time required to recognize 358 keywords in FreeOCR is greater than Time required for Asprise OCR with higher resolution. Accuracy of Recognized Keyword from Plain Text Page with 300 dpi resolution is 99% in Asprise OCR SDK and 98% in FreeOCR that means keyword recognition rate with good resolution is nearly same. Plain Text Page with 150 dpi resolution (low resolution) is better in Asprise OCR SDK than Free OCR. This shows the efficacy of Asprise OCR for low resolution images as well. Text layouts on input documents are preserved in Asprise OCR SDK. Besides characters (letters and numbers), Asprise OCR can recognize almost every kind of bar code.

References

[1] Sanjay Kumar, Narendra Sahu, Aakash Deep, Khushboo Gavel, Miss Rumi Ghosh, "Offline Handwriting Character Recognition (for use of medical purpose) Using Neural Network", International Journal of Engineering and Computer Science, Volume 5 Issue 10, Page No. 18612-18615, Oct. 2016.

[2] Paul D. Thouin, Chein-I Chang, "A method for restoration of low-resolution document image" , International Journal on Document Analysis and Recognition, Springer-Verlag, November 2000.

[3] Chao Dong, Ximei Zhu, Yubin Deng, Chen Change Loy, "Boosting Optical Character Recognition: A Super Resolution Approach", ICDAR 2015 competition on text image super-resolution, 7 June 2015.

[4] Imad Qasim Habeeb, Shahrul Azmi Mohd Yusof, Faudziah, B. Ahmad, "Improving Optical Character Recognition Process for Low Resolution Images", International Journal of Advancements in Computing Technology (IJACT) Volume 6, Number 3, May 2014.

[5] Najib Ali Mohamed Isheawy, Habibul Hasan, "Optical Character Recognition (OCR) System", IOSR Journal of Computer Engineering (IOSR-JCE) Volume 17, Issue 2, Ver. II , PP 22-26 (Mar – Apr. 2015).

[6] M Swamy Das,Ram Mohan Rao Kovvur, "Evaluation of Neural Based Feature Extraction Methods for Printed Telugu OCR System", Advances in Computer Science and Information Technology (ACSIT) ,Volume 2, Number 11, pp. 85-90, April- June, 2015.

[7] Chirag Patel, Atul Patel, Dharmendra Patel, "Optical Character Recognition by Open Source OCR Tool Tesseract: A Case Study", International Journal of Computer Applications (0975 – 8887) Volume 55– No.10, October 2012.

[8] FreeOCR Url [Online].Available: http://www.paperfile.net/

[9] Asprise OCR Url [Online] Available: https://en.wikipedia.org/wiki/Asprise_OCR