Which OCR toolset is good and why? A comparative study

Pooja Jain[1,*], Dr. Kavita Taneja[1], Dr. Harmunish Taneja[2]

[1]Dept. Of Computer Science & Applications, Panjab University, Chandigarh, India.

[2]Dept. Of Computer Science & Information Tech., DAV College, Sec - 10, Chandigarh, India.

**Corresponding author: poojajain9199@gmail.com*

Abstract

Optical Character Recognition (OCR) is a very active research area in many challenging fields like pattern recognition, natural language processing (NLP), computer vision, biomedical informatics, machine learning (ML), and artificial intelligence (AI). This computational technology extracts the text in an editable format (MS Word/Excel, text files, etc.) from PDF files, scanned or hand-written documents, images (photographs, advertisements, and alike), etc. for further processing and has been utilized in many real-world applications including banking, education, insurance, finance, healthcare and keyword-based search in documents, etc. Many OCR toolsets are available under various categories, including open-source, proprietary, and online services. This research paper provides a comparative study of various OCR toolsets considering a variety of parameters.

Keywords: ABBYY FineReader; Calamari; Google Docs; OCR; Tesseract.

1. Introduction

OCR (Bokser, 1992; Mori *et al.*, 1992) is a commonly used technology for recognizing text within digital images such as scanned documents, advertisements, photographs, etc. It is widely used as an information entry tool that can extract useful information from scanned documents, including printed forms (filled by users), computerized receipts, bank statements, invoices, business cards, passport documents, mails, or any other suitable documentation. Other applications include searching within institutional repositories and scanned legal documents, automatic number plate recognition (ANPR), processing cheques in banks, recognizing barcodes, testing text-based captcha codes, etc.

1.1. OCR process

OCR process (Goswami *et al.*, 2013; Cao, 2014; Tomaschek, 2018) generally goes through multiple stages, as shown in Figure 1, including Image-acquisition **(**downloading image from an online source or capturing it using a camera or scanner), Pre-processing (modifying image in a way that may increase the accuracy of OCR), Binarization (separating the content from the background), Layout Analysis (a division of the document into various homogeneous regions), Character level segmentation (segmentation of the image into lines, words, and characters), Recognition (feature extraction of every character image) and finally, Classification (determining the output characters) followed by Postprocessing where classification results can be enhanced using various language models and dictionaries.

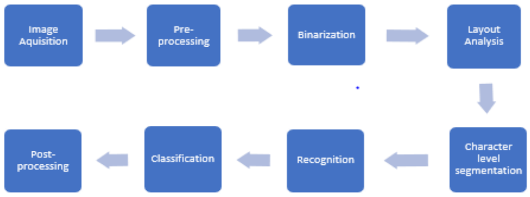

Fig. 1. OCR Process

1.2. OCR Challenges

OCR faces many problems in recognizing printed or handwritten characters such as image deformation (disconnected line segments, isolated dots, breaks or holes in lines, rotation of text, etc.), shape discrimination (some characters have very similar shapes like 0 (zero) and O, 5 and S, 2 and Z, 6 and G, etc.), size and pitch variations, cluttered background, a camera captured documents (motion and out-of-focus blur), multilingual documents, text formatting, and complicated structures and natural scene text (variation in illumination conditions and fonts, etc.). Touching characters and different typefaces used in early printed books (15th-19th century) pose additional text recognition difficulties. Recognition of calligraphy typefaces and fonts is another challenge (Al-Hmouz, 2020).

1.3. OCR Toolsets

OCR toolsets are software that focus on accurate character recognition in a reliable manner. OCR toolsets can be broadly classified into three categories:

(i) Proprietary
(ii) Open Source
(iii) Online

Figure 2 presents various popular OCR toolsets in different categories. Since many proprietaries, open-source, and online OCR toolsets with varied features are available to choose from, this research paper reviews and analyse the performance of various OCR toolsets and provide better insight to the OCR study with an aim to help researchers in making a right choice of an OCR toolset specific to their application domain. The rest of the paper is arranged as follows. First proprietary OCR toolsets are introduced, followed by open-source and online OCR toolsets. Then a literature review for various comparative studies conducted for OCR toolsets is presented, followed by the discussion and conclusion of the study.

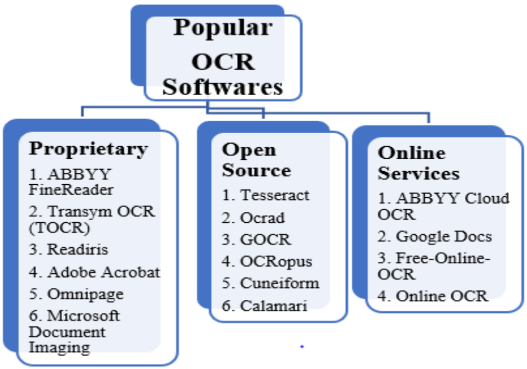

Fig. 2. Popular OCR Softwares.

2. Proprietary OCR Toolsets

Proprietary OCR toolsets are usually paid for and supported by developers. They generally have an excellent graphical user interface (GUI). Popular proprietary OCR toolsets are discussed in this section, and their important features are summarized in Table 1.

2.1. ABBYY FineReader

ABBYY FineReader achieves up to 100 % word-level accuracy with high quality images in English language (*https://abbyy.technology/_media/en:products:fre:win:v11:frengine11_performance_guide.pdf*), and has been an undisputed choice for layout analysis and OCR. It can be accessed in two different ways: ABBYY FineReader SDK and ABBYY Online/ cloud service.

2.2. Transym OCR (TOCR) (*http://www.transym.com/tocr-the-integrators-choice.htm*)

TOCR (Tafti *et al.*, 2016) is specifically designed keeping in mind the ease of integration with other softwares. With a very light and efficient GUI, it is claimed that Transym has the capability to read blurred, obscure, and even broken characters (Vithlani *et al.*, 2015).

2.3. Readiris (*https://www.irislink.com/EN-US/c1729/Readiris-17--the-PDF-and-OCR-solution-for-Windows-.aspx*)

Preserving the original page layout, Readiris automatically converts text from PDF files, images, or paper documents into fully editable files. Being compatible with most of the scanners in the market, it supports many different input formats and provides an attractive and intuitive GUI.

2.4. Adobe Acrobat (*https://acrobat.adobe.com/in/en/acrobat/how-to/ocr-software-convert-pdf-to-text.html*)

Adobe Acrobat automatically converts image files, scanned documents, and PDF files into searchable/editable documents while preserving the format and its accuracy is reportedly high. It provides fewer language options comparing with ABBYY FineReader, but it is more pervasive software as it is more business-oriented and less academic.

2.5. OmniPage (*https://www.kofax.com/Products/omnipage*)

OmniPage (https://www.nuancesoftwarestore.com/omnipage-ultimate) claims 99% or more character-accuracy and converts paper documents and PDFs into fully editable digital files preserving all type of formatting. Using OmniPage, digital camera photos and other images can also be converted into text files.

2.6. Microsoft Office Document Imaging (MODI) (*https://support.microsoft.com/da-dk/help/982760/install-modi-for-use-with-microsoft-office-2010*)

MODI is a free OCR tool that can scan hard copies of documents and import them to MS Word for editing. By default, MODI can perform OCR in three languages (English, French, and Spanish). For other languages, their language pack must be installed first. However, multiple languages cannot be set for a single document in MODI.

3. Open-source OCR Toolsets

Open-source OCR engines are best controlled via their command-line interfaces, and most of them do not have a GUI. Some of the popular open-source OCR toolsets are discussed in this section, and their important features are summarized in Table 2.

3.1. Tesseract

Tesseract (Smith, 2007; Patel *et al.*, 2012) comes with an easy-to-use command-line tool called 'tesseract.' It can be integrated in C++ or Python code by using Tesseract's API. Many GUI desktop applications use Tesseract as a text recognition engine, including FreeOCR, OCRFeeder, PDF OCR X, QTesseract, YAGF, gImageReader, Lector, VietOCR, SunnyPage, and Lime OCR. Also, WeOCR, CustomOCR, i2OCR, and NewOCR are web applications using the Tesseract tool (Vithlani *et al.*, 2015). Its latest version, Tesseract4.0 (*https://github.com/tesseract-ocr/tesseract/wiki/4.0-with-LSTM*), uses a kind of Recurrent Neural Network (RNN) called Long Short Term Memory (LSTM) based recognition engine.

3.2. Ocrad (*https://www.gnu.org/software/ocrad/manual/ocrad_manual.html*)

Ocrad supports narrowing down the character search by defining which character sets to recognize. Ocrad recognizes characters very fast, but at the same time, it is very sensitive to character defects, and it is difficult to modify Ocrad to recognize new characters. Best character recognition results are achieved when characters are at least 20 pixels high, or the image is scanned at 300 dpi.

3.3. GOCR (*http://jocr.sourceforge.net/*)

GOCR can be used either as a stand-alone console application or as a back-end (OCR engine) to other programs. Written in C language, its recognition process takes two passes. In the first pass, the entire document is called. In the second pass, the unknown characters are called (Dhiman *et al.*, 2013). It is claimed that GOCR can handle single-column sans-serif fonts, which are 20–60 pixels in height. Trouble is reported with serif fonts, italic fonts, slanted fonts, small fonts, heterogeneous fonts, coloured images, handwritten text, overlapping characters, large angles of skew, noisy images, multiple columns, tables, complicated layouts, and text other than Latin alphabets.

3.4. OCRopus (*https://code.google.com/p/ocropus/*)

Supporting a command-line based user interface, OCRopus (Breuel, 2008) has a very modular design giving the flexibility to perform each step of OCR (e.g., binarization, page layout analysis, text line recognition, etc.) individually using separate commands and the user can use different modules of his/her own choice to perform these tasks. Its latest version, OCRopus3 (Breuel *et al.*, 2013), uses bi-directional LSTM models using PyTorch networks.

3.5. CuneiForm (*https://www.softpedia.com/get/Office-tools/Other-OfficeTools/CuneiForm.shtml*)

CuneiForm(*https://en.wikipedia.org/wiki/CuneiForm_(software)*) can be used from the command line (as a stand-alone application) or with other programs (as a background application). It uses language dictionaries to improve recognition results. English is the default recognition language, but the user can choose other languages as desired.

3.6. Calamari (*https://github.com/Calamari-OCR*)

Calamari (Reul *et al.*, 2019) is one of the latest open-source OCR line recognition tool. Based on state-of-the-art Deep Neural Networks (DNNs) implemented in Tensorflow (including Convolutional Neural Networks (CNN) and LSTM layers), it uses techniques such as pretraining and voting, which help in minimizing its character error rates (CERs). Calamari doesn't offer a full OCR pipeline and just focuses on recognizing text from text line images. It can be integrated in existing OCR pipelines and can replace their OCR-engine efficiently.

4. OCR Online services

Using OCR online services, there is no need to download or install any OCR software. The user is only required to upload the input file, select the language(s), output format (optionally), and the output is generated. Few important OCR online services are discussed in this section, and their important features are summarized in
Table 3.

4.1. ABBYY Cloud OCR (*https://www.abbyy.com/en-eu/cloud-ocr-sdk/*)

Running on Microsoft Azure infrastructure, ABBYY's Cloud OCR SDK is a Web OCR service that provides excellent text recognition quality. This web service can be easily integrated into your own applications using a Web API. Converted files can be exported to Google Docs, Drop Box, and Ever note (Vithlani *et al.*, 2015).

4.2. Google Docs (*http://docs.google.com*)

Google Docs is primarily a cloud document storage and editing platform offered by Google within the Google Drive service (http://drive.google.com). Once an image or a PDF file is uploaded to Google Drive, OCR conversion can be performed by right clicking on the file and selecting the option "Open with Google Docs." The extracted text is in editable form, which can be downloaded.

Table 1. Comparison of Important Proprietary OCR Toolsets.

S.No.	Proprietary OCR Toolset	Available Online	Multilingual support	Operating System	Input Formats supported	Output Formats supported	Important Features
1.	ABBYY FineReader	Yes	Yes (190+)	Windows, MAC OS X, Linux	PDF, BMP, TIFF, GIF, PNG, JPEG, PCX, DCX, DjVu, JBIG2, WDP	DOC(X), XLS(X), PPT(X), RTF, XML, PDF, CSV, TXT, ALTO, FB2, DBF, EPUB, HTML, ODT	• Pre-processing techniques include noise removal, skew correction, straightening text lines, trapezoidal distortion correction. • Uses AI and ML for precise document reconstruction and higher accuracy.
2.	Transym	No	Yes (11)	Windows	PDF, BMP, TIFF	TXT, RTF	• Automatically detect the page or image orientation. • Can identify text with background defects (extremely light or dark backgrounds, deformation, and speckle). • Uses lexicon for maximising word accuracies and reliability.
3.	Readiris	No	Yes (138)	Windows, MAC OS X, iOS, Android	PDF, JPEG, PNG, BMP, TIFF, JBIG2, JPEG2000	PDF, PDF/A, HTML, XML, RTF, TXT, ODT, WordML, SpreadsheetML, CSV, DOC(X), XLS(X), XPS, ePub	• Pre-processing techniques include adaptive binarization, de-speckle filters, de-skew feature, document rotation, dark border removal, line removal, and colour dropout. • Font-independent text recognition is complemented by self-learning techniques derived from proprietary neural networks. • Uses proprietary dictionaries.
4.	Adobe Acrobat	No	Yes	Windows, MAC OS X, iOS, Android	PDF, BMP, JPG/ JPEG, GIF, TIF/ TIFF, PNG, PCX, RLE, DIB	DOC(X), XLSX, PPTX, HTML, RTF, PS, EPS, XML, Edit text in PDF, TXT, CSV	• Allows editing of text in PDFs. • Automatically generates a custom font (for adding or editing within the PDF document) that looks like the same font as in the original document. • Smart PDFs can be created (only searching and copying capabilities without editing).
5.	OmniPage	Yes	Yes (120+)	Windows, MAC OS X, Linux	BMP, DCX, GIF, JBG, JP2, MAX, PCX, PDF, XIF, XPS	DOC(X), XLS(X), PPT(X), RTF, MP3, ePUB, XML, PDF, PDF/A, Searchable PDF, Corel WordPerfect, HTML Text	• Pre-processing tools and de-speckling methods are available to reduce background noise and enhancement of text and diagrams. • Removal of punch holes and auto-cropping of margins. • Text extraction from shaded and coloured documents with very little human intervention. • Able to perform document reading through mobile devices that support MP3 audio files.
6.	Microsoft Office Document Imaging	No	Yes (by-default 3)	Windows	TIF/TIFF, MDI	DOC/DOCX, MDI, TIFF	• De-skews and re-orients the page where required. • Produces TIFF files that violate the TIFF standard specifications and are only usable by MODI.

Note: All proprietary OCR softwares discussed above have GUI and are not free except MODI, which is free.

Table 2. Comparison of Important Open-source OCR Toolset

S.No.	**Open-source OCR Toolset**	**Available Online**	**Multilingual support**	**Operating System**	**Input Formats supported**	**Output Formats supported**	**Important Features**
1.	Tesseract4.0	No	Yes (100+)	Windows, MAC OS X, Linux, Android	TIFF, JPEG, JFIF, PNG, PNM (PGM, PBM, PPM), BMP	TXT, PDF, hOCR	• Pre-processing techniques include orientation detection and minor skew correction. • Can use multiple languages in a single scan. • Machine learning support for recognizing new languages, symbols, and fonts. • No GPU support till date.
2.	Ocrad	Yes Ocrad.js	Yes (Latin alphabets)	MAC OS X, Linux, BSD	PNM (PGM, PBM, PPM)	TXT	• Pre-processing transformations including cut, rotate, scale, and layout detection. • Both in-built and user-defined filters can be used for the post-processing step.
3.	GOCR	Yes GOCR.js	Yes (20+)	Windows, MAC OS X, Linux, BSD	PNM (PGM, PBM, PPM), some PCX and TGA formats	Text file	• GUI (gocr.tcl). • No training data is required (no neural network) or large font bases to store. • Barcodes can also be recognized and translated.
4.	OCRopus	No	Yes (Languages with Latin script)	MAC OS X, Linux, BSD	PNG	TXT, hOCR, PDF, HTML	• Can be trained to recognize new languages or different fonts. • Used for Google Books. • GPU support in OCRopus3.
5.	Cuneiform	No	Yes (25+)	Windows, MAC OS X, Linux, BSD	PNG, BMP, JPG	HTML, hOCR, RTF, TeX, TXT	• GUI for Windows, command based for Linux. • Saves text formatting and recognizes complicated tables of any structure. • A mixture of Russian and English can also be recognized.
6.	Calamari	No	Yes	**Not Known**	PNG, JPG, H5	GT.TXT, XML, ABBYY.XML, HDF5	• Uses Cross Fold Voting, Data Augmentation, Pretraining. • GPU support.

Table 3. Comparison of Important Online OCR Toolsets.

S.No	**Online OCR Toolsets**	**Multilingual support**	**Free**	**operating System**	**Input Formats supported**	**Output Formats supported**	**Important Features**
1.	ABBYY Web service	Yes (190+)	No (Free trial version is available)	Platform independent	PDF, BMP, TIFF, GIF, PNG, JPEG, PCX, DCX, DjVu, JBIG2, WDP	DOC(X), XLS(X), PPT(X), RTF, XML, PDF, CSV, TXT, ALTO, FB2, DBF, EPUB, HTML, ODT	• Provides highest data security by complying with the relevant data protection laws. • Preserves formatting. • Multipage documents can also be converted. • Maximum input file size: 30 MB. • For a multilingual document, up to 3 recognition languages can be chosen.
2.	Google Docs	Yes (83+)	Yes	Windows, Android, MAC OS X, iOS, BlackBerry, ChromeOS	PDF, PNG, JPG, GIF, TIFF	DOC(X), DOCM, DOT(X), DOTM, HTML, TXT, RTF, ODT	• Automatically determine the language of the document, and no need to specify the language. • Currently, OCR works best on cleanly scanned, high-resolution documents in the most commonly used typefaces. • Maximum input file size: 50 MB.
3.	Free-Online OCR	No (English only)	Yes	Browser-Based	GIF, JPG, BMP, PNG, TIF, PDF	DOC, RTF, PDF, TXT	• Automatically rotates pages. • Supports low-resolution images. • Preserves the original layout and formatting. • Maximum input file size: 200 MB.
4.	Online OCR	Yes (46)	Yes	Browser-Based	JPEG/JPG, PNG, PDF, BMP, TIF/TIFF, PCX, GIF, ZIP	DOCX, XLSX, TXT, HTML, PDF, RTF	• For best text recognition, input images should have a resolution of 200-400 DPI. • Maximum input file size: 200 MB • Automatically rotates images (full-page de-skew) for better recognition. • Non-text, coloured regions are reinserted into the output document.

Note: All Online OCR Toolsets discussed above have GUI.

4.3. Free-Online-OCR (*http://www.free-online-ocr.com/*)

Free-Online-OCR can achieve high recognition accuracy even with low-quality documents, including screenshots and faxes. The accuracy is further increased with the help of an integrated dictionary.

4.4. Online OCR (*http://www.onlineocr.net/service/about*)

Online OCR can convert digital camera-captured images, photographs, faxes, and scanned documents into various searchable and editable formats. Documents written in more than one language can also be processed. It allows free conversion of 15 images per hour in a guest mode without registration, whereas free Registration provides extra features.

5. Review of comparative studies of popular OCR softwares

Many experimental studies have been conducted to compare and analyse the performance of various OCR toolsets on standard as well as non-standard datasets.

Dhiman *et al.*, 2013, used different parameters such as image type, font type, brightness, and resolution and compared Tesseract and GoCR based on precision as well as accuracy. The authors concluded that Tesseract outperforms GoCR in most of cases.

Gabasio, 2013, calculated the mean error values of various proprietary (TOCR, Leadtools, ABBYY, OCR API Service) and open-source (OCRopus, Tesseract, CuneiForm, Ocrad) OCR tools and concluded that the mean error rates of tested proprietary OCR toolsets are much lower than popular open-source OCR toolsets and suggested to invest in proprietary OCR toolsets if the scanned images are of different quality. Among the open-source category, the mean error rates of OCRopus and Tesseract were comparable, but OCRopus takes a lot more time in conversion, and hence, Tesseract becomes a better choice.

Patel *et al.*, 2012, compared the performance of open-source Tesseract with proprietary Transym for ANPR. It is observed that Tesseract provides better accuracy than Transym for both grayscale and coloured images. Also, Tesseract was found to be faster than Transym. The standard deviation (for accuracy) of Tesseract was also less than Transym.

Tomaschek, 2018, conducted a comparative study of popular open-source OCR tools, including Tesseract3, Tesseract4.0, GOCR, Ocrad, and established proprietary OCR tools, including Nuance OmniPage and Readiris16 on a prepared dataset which contained a balanced mix of various document classes. The results clearly indicate that proprietary Nuance OmniPage could achieve 100% accuracy, whereas none of the open-source tools could achieve 100% accuracy. Also, in the open-source category, Tesseract (both version 3 & 4.0) performs better than GOCR and Ocrad.

In another test conducted by Tomaschek, 2018, on a synthetic page with different open source (Tesseract3, Tesseract4.0, GOCR, Ocrad, and OCRopus) and proprietary OCR tools (Cuneiform, Nuance OmniPage, and Readiris16), it was noted that proprietary Cuneiform and Nuance OmniPage could achieve 100% word- accuracy whereas in open-source category only Tesseract4.0 could achieve 100% word- accuracy ratio. While comparing the time taken by the tested OCR tools, OCRopus was found to be slowest and Tesseract4.0 being the fastest among these.

Tafti *et al.*, 2016, performed the experimental evaluation of four popular OCR toolsets (Google Docs, ABBYY FineReader, Tesseract, and Transym) using a dataset of images from different categories. It is found that Google Docs and ABBYY FineReader performed more consistently across different image categories with a population standard deviation of 18.19 and 18.02, respectively, as compared to 25.56 and 25.79 of Tesseract and Transym, respectively.

Vijayarani *et al.*, 2015, evaluated the performance of eight OCR tools, including Google Docs, Free OCR to Word Convert, i2OCR, FreeOCR, Convertimagetotext.net, OCR Convert, Free Online OCR, and Online OCR on a sample input image. It is concluded that all the tested OCR tools (except Free OCR to Word Convert) performed reasonably good while converting characters from the text images, but none of the tested OCR tools performed satisfactorily while converting mathematical symbols and equations.

Asad *et al.,* 2016, compared the performance of three popular OCR toolsets for camera-captured blurred documents. The experimental evaluation on the SmartDoc-QA dataset shows that out of ABBYY FineReader, Tesseract, and OCRopus, the lowest CER of 38.9% is achieved by ABBYY FineReader. It is further stated that the performance of these OCR toolsets is limited by the binarization techniques employed or by the factor of segmentation. A new framework called BLSTM is proposed, which overcomes the problem of segmentation-based OCR systems and eliminates the need for binarization of blurred documents.

Reul *et al.*, 2017, concluded that both ABBYY and Tesseract don’t yield satisfactory results for early printed books. On the other hand, OCRopus3, when aided with text analysis tools like Aletheia (manual segmentation) or fully automated open-source tool: LAREX (Layout Analysis and Region Extraction) to perform segmentation, provided high-quality OCR result with over 97% character-accuracy and around 92% word-accuracy on an early printed book (15th century) within a reasonable amount of time. The results clearly show that after being thoroughly trained, OCRopus could recognize even the earliest printed typefaces.

Borisyuk *et al.*, 2018, presented a scalable OCR system Rosetta which is used to extract text from a huge volume of images uploaded to Facebook and Instagram every day and facilitates many applications like search and recommendation of images. Rosetta's OCR process is based on Faster-RCNN for detecting text containing regions of the image, followed by a fully-convolutional character-based recognition model for recognizing the text in those locations. Scene text or photographs may include email-ids, special symbols, URLs and words from different languages and hence the use of pre-defined dictionaries may not suit here. Therefore, instead of using a dictionary-based recognition model, Rosetta uses a character-based recognition model.

Reul *et al.*, 2019, compared Calamari with other popular open-source OCR tools on the UW3 (University of Washington III) dataset, and it is found that Calamari yields superior recognition capabilities with 0.114% CER as compared to OCRopus3 (0.436%) and Tesseract4.0 (0.397%). It is also observed that Calamari and OCRopus3 support batched GPU training and prediction and hence, are faster than Tesseract4.0.

Namysl *et al.*, 2019, presented a robust and fast, deep learning-based multi-font OCR engine that uses a segmentation-free text recognition method and a novel data augmentation technique resulting in improved text recognition capabilities. A comparison of the presented OCR engine with leading OCR tools including Tesseract3, Tesseract4.0, ABBYY FineReader, and OmniPage revealed that better recognition results were obtained by the presented OCR engine on distorted documents with background textures within comparable execution time. The low recognition performance by the compared tools for distorted inputs is due to their inability to segment the characters in noisy backgrounds adequately.

6. Discussion

With up to 100% word-level accuracy in high quality images and covering more than 190 languages, proprietary ABBYY FineReader clearly defines the state-of-the-art for modern prints. However, for early printed books, OCRopus3 provides high-quality OCR results within a reasonable amount of time.

Both OCRopus3 and Tesseract4.0 have implemented LSTM based recognition engine for better recognition capabilities. However, the major disadvantage of OCRopus is that it is not available for Windows operating systems, whereas Tesseract is available for all three major operating systems (Windows/ Linux/ MAC OS). Another disadvantage of OCRopus is that it needs a set of commands in sequence for complete OCR rather than a single command. Though this modularity has its own advantages but it makes OCRopus a slow tool.

For modern English prints, among open source categories, Calamari yields better recognition results as compared to OCRopus3 and Tesseract4.0 (Reul *et al.*, 2019), and its GPU support helps in high speed training and recognition. However, Calamari is not designed as a complete OCR solution and focuses solely on text recognition, whereas other open-source OCR tools like Tesseract4.0 are designed to support the full pipeline of OCR from image to text. Also, like ABBYY, Tesseract4.0 uses dictionaries and language modelling, whereas Calamari and OCRopus do not use these.

For extracting text from a large volume of images uploaded to Facebook and Instagram every day, a scalable OCR toolset named Rosetta (Borisyuk *et al.*, 2018) is used, which uses a fully CNN based text recognition model instead of recurrent LSTM used by other popular OCR toolsets including Tesseract4.0 and OCRopus3. It costs a small loss in the accuracy, but the inference time is reduced, the desired feature in many applications like quick search and recommendation of images from a large dataset. Rosetta does not use pre-defined language dictionaries and instead uses a character-based recognition model to facilitate the recognition of email-ids, special symbols, URLs, and words from different languages in a single image.

When it comes to text recognition in distorted documents with background textures, the leading OCR toolsets, including Tesseract4.0, ABBYY, and OmniPage, do not give satisfactory results. A new OCR tool is presented by Namysl *et al.*, 2019, which performs much better than the established OCR tools for distorted inputs within comparable execution time and is able to recognize superscripts, subscripts as well as different and alternating font styles located on the same page better than the leading OCR toolsets.

OCR online services are generally free, very useful, and convenient to use. However, uploading the file on the internet to their servers may have some privacy and security concerns. Also, if the document is big, the user may face time/bandwidth issues. Most online services limit the file size which can be uploaded or the no. of pages that can be processed daily/weekly for free. For bigger jobs, the user needs to pay.

7. Conclusion

OCR is a challenging research area that deals with various complexities, including image quality, background noise, skewed or speckled images, recognition languages, different typefaces and fonts, various input-output formats, text formatting and complex structures, natural scene text, etc. Lots of OCR toolsets are readily available for use in various domains. This paper provided a brief introduction and important features of various popular OCR toolsets in different categories, including open-source, proprietary, and online categories. A review of various comparative studies conducted to evaluate the performance of these popular OCR tools has been presented here. It is concluded that different OCR tools have different capabilities, and a single OCR toolset may not fit in all the domains. Image quality plays an important role in text recognition. For high-quality images, proprietary ABBYY is best-in-class OCR software. For books printed later than 19th century, ABBYY gives best results, but for early printed books, OCRopus3 provides much better OCR results. For modern English prints, in the open-source category, Calamari yields better recognition results as compared to OCRopus3 and Tesseract4.0, But when a single software is required which can perform the complete pipeline of OCR in one go, Tesseract4.0 is the most popular choice among the open-source category. When dealing with a large volume of images, Rossetta provides faster recognition results and facilitates natural scene text recognition. For distorted inputs with noisy backgrounds, a new lexicon-free OCR toolset is presented by Namysl *et al.*, 2019, which provides much better text recognition results as compared to established OCR toolsets. Many free online OCR services are available, including Google Docs, Online-OCR, Free-Online-OCR, etc., which allow the users to convert images into text files without downloading the OCR toolsets on their machines, but file security may be a concern using these online services.

References

Al-Hmouz, R. (2020) Deep learning autoencoder approach: Automatic recognition of artistic Arabic calligraphy types. Kuwait Journal of Science, **47**(3).

Asad, F.; Ul-Hasan, A.; Shafait, F. & Dengel, A. (2016) High-Performance OCR for Camera-Captured Blurred Documents with LSTM Networks. In *12th IAPR Workshop on Document Analysis Systems (DAS)* (pp. **7-12**). IEEE.

Bokser, M. (1992) Omnidocument technologies. Proceedings of the IEEE, **80**(7), pp.**1066-1078**.

Borisyuk, F.; Gordo, A. and Sivakumar, V. (2018) Rosetta: Large scale system for text detection and recognition in images. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp. **71-79**). ACM.

Breuel, T.M. (2008) The OCRopus open-source OCR system. In *Document Recognition and Retrieval XV* (Vol. **6815**, p. **68150F**). International Society for Optics and Photonics.

Breuel, T.M.; Ul-Hasan, A.; Al-Azawi, M.A. & Shafait, F. (2013) High-performance OCR for Printed English and Fraktur using LSTM networks. In *12th International Conference on Document Analysis and Recognition* (pp. **683-687**). IEEE.

Cao, H. (2014) Machine-printed character recognition. Handbook of Document Image Processing and Recognition: **331-358**.

Dhiman, S. and Singh, A. (2013) Tesseract vs. gocr a comparative study. International Journal of Recent Technology and Engineering, **2**(4), p.**80**.

Gabasio, A. (2013) Comparison of optical character recognition (OCR) software. Master’s thesis Lund University, Sweden. pp. **8-19**.

http://fileadmin.cs.lth.se/intern/Utskrifter/2013-10%20Rapport.pdf

Goswami, R. & Sharma, O.P. (2013) A Review on Character Recognition Techniques. International Journal of Computer Applications **83**(7).

Mori, S.; Suen, C.Y. & Yamamoto, K. (1992) Historical review of OCR research and development. Proceedings of the IEEE, **80**(7), pp.**1029-1058**.

Namysl, M. & Konya, I. (2019) Efficient, Lexicon-Free OCR using Deep Learning. In *International Conference on*

Document Analysis and Recognition (ICDAR) (pp. **295-301**). IEEE.

Patel, C.; Patel, A. & Patel, D. (2012) Optical character recognition by open-source OCR tool tesseract: A case study. International Journal of Computer Applications, **55**(10), pp.**50-56**.

Reul, C.; Dittrich, M. and Gruner, M. (2017) Case Study of a highly automated Layout Analysis and OCR of an incunabulum:'Der Heiligen Leben'(1488). In *Proceedings of the 2nd International Conference on Digital Access to Textual Cultural Heritage* (pp. **155-160**). ACM.

Reul, C.; Christ, D.; Hartelt, A.; Balbach, N.; Wehner, M.; Springmann, U.; Wick, C.; Grundig, C.; Büttner, A. & Puppe, F. (2019) OCR4all-an open-source tool providing a (semi-) automatic OCR workflow for historical printings. Applied Sciences, **9**(22), pp.**1-30**.

Smith, R. (2007) An overview of the Tesseract OCR engine. In *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)* (Vol. 2, pp. **629-633**). IEEE.

Tafti, A.P.; Baghaie, A.; Assefi, M.; Arabnia, H.R.; Yu, Z. & Peissig, P. (2016) OCR as a service: an experimental evaluation of Google Docs OCR, Tesseract, ABBYY FineReader, and Transym. *International Symposium on Visual Computing*. pp. **735-746**. Springer, Cham.

Tomaschek, M. (2018) Evaluation of off-the-shelf OCR technologies. Bachelor thesis Masaryk University, Brno, Czech Republic. pp. **3-24.**

https://is.muni.cz/th/v09x6/

Vijayarani, S. & Sakila, A. (2015) Performance comparison of OCR tools. International Journal of UbiComp (IJU), **6**(3), pp.**19-30**.

Vithlani, P. & Kumbharana, C.K. (2015) Comparative Study of Character Recognition Tools. International Journal of Computer Applications **118**(9).

Submitted **:** 20/04/2020

Revised **:** 20/08/2020

Accepted **:** 24/08/2020

DOI **:** 10.48129/kjs.v48i2.9589