OCR as a Service: An Experimental Evaluation of Google Docs OCR, Tesseract, ABBYY FineReader, and Transym

Article · December 2016

Mehdi Assefi
University of Georgia

OCR as a Service: An Experimental Evaluation of Google Docs OCR, Tesseract, ABBYY FineReader, and Transym

Ahmad P. Tafti[1(✉)], Ahmadreza Baghaie[2], Mehdi Assefi[3], Hamid R. Arabnia[3], Zeyun Yu[4], and Peggy Peissig[1(✉)]

[1] Biomedical Informatics Research Center, Marshfield Clinic Research Foundation, Marshfield, WI 54449, USA
{pahlavantafti.ahmad,peissig.peggy}@mcrf.mfldclin.edu

[2] Department of Electrical Engineering, University of Wisconsin-Milwaukee, Milwaukee, WI 53211, USA

[3] Department of Computer Science, University of Georgia, Athens, GA 30602, USA

[4] Department of Computer Science, University of Wisconsin-Milwaukee, Milwaukee, WI 53211, USA

Abstract. Optical character recognition (OCR) as a classic machine learning challenge has been a longstanding topic in a variety of applications in healthcare, education, insurance, and legal industries to convert different types of electronic documents, such as scanned documents, digital images, and PDF files into fully editable and searchable text data. The rapid generation of digital images on a daily basis prioritizes OCR as an imperative and foundational tool for data analysis. With the help of OCR systems, we have been able to save a reasonable amount of effort in creating, processing, and saving electronic documents, adapting them to different purposes. A set of different OCR platforms are now available which, aside from lending theoretical contributions to other practical fields, have demonstrated successful applications in real-world problems. In this work, several qualitative and quantitative experimental evaluations have been performed using four well-know OCR services, including Google Docs OCR, Tesseract, ABBYY FineReader, and Transym. We analyze the accuracy and reliability of the OCR packages employing a dataset including 1227 images from 15 different categories. Furthermore, we review the state-of-the-art OCR applications in healtcare informatics. The present evaluation is expected to advance OCR research, providing new insights and consideration to the research area, and assist researchers to determine which service is ideal for optical character recognition in an accurate and efficient manner.

1 Introduction

Optical character recognition (OCR) has been a very practical research area in many scientific disciplines, including machine learning [1–3], computer vision [4–6], natural language processing (NLP) [7–9], and biomedical informatics [10–12]. This computational technology has been utilized in converting scanned,

G. Bebis et al. (Eds.): ISVC 2016, Part I, LNCS 10072, pp. 735–746, 2016.
DOI: 10.1007/978-3-319-50835-1_66

hand-written, or PDF files into an editable text format (e.g., text file or MS Word/Excel file) for further processing tasks [13,14]. OCR has contributed to significant process improvement in many different real world applications in healthcare, finance, insurance, and education. For example, in healthcare there has been a need to deal with vast amounts of patient forms (e.g., insurance forms). In order to analyze the information in such forms, it is critical to input the patient data in a standarized format into a database so it can be accessed later for analysis. Using OCR systems, we are able to automatically extract information from the forms and enter it into databases, so that every patient's data is immediately recorded. OCR really simplifies the process by turning those documents into easily editable and searchable text data. In the sense of software engineering "Software as a Service" (SaaS), as an architectural model behind the centralized computing, has emerged as a design pattern and also a delivery model in which a software could be accessed by both human-oriented and application-oriented standards [15–19]. Human users can get the SaaS system through a web browser, and an application will utilize the service using APIs (application programming interfaces).

To date, several attempts have been made to design and develop OCR services and/or packages, such as Google Docs OCR [20], Tesseract [21,22], ABBYY FineReader [23,24], Transym [25], Online OCR [26], and Free OCR [27]. Based on core functionalities, including recognition accuracy, performance, multilingual support, open-source implementation, delivery as a software development kit (SDK), high availability, and rating in the OCR community [28,29], the present contribution is mainly focused on the experimental evaluation of Google Docs OCR, Tesseract, ABBYY FineReader, and Transym. The current work is expected to provide better insights to the OCR study, and address several capabilities for possible future enhancements.

The rest of the paper is arranged as follows. The Google Docs OCR, Tesseract, ABBYY FineReader, and Transym OCR systems will be introduced in Sect. 2. In Sect. 3 we review, from an application perspective, the state-of-the-art OCR systems in healthcare informatics. Experimental validations including the dataset, testbed, and the results will be reported in Sect. 4. Section 5 provides discussion and concludes the work.

2 OCR Toolsets

OCR toolsets and their underlying algorithms not only focus on text and character recognition in a reliable manner, but may also address: (1) Layout analysis in which they can detect and understand different items in an image (e.g., text, tables, barcodes), (2) Support of various alphabets, including English, Greek, Persian, and etc., and (3) Support of different types of input images (e.g., TIFF, JPEG, PNG, PDF) and capabilities to export text data in different output formats. The basis of OCR methods dates back to 1914 when Goldberg designed a machine that was able to read characters and turn them into standard telegraph code [13]. With the emergence of computerized systems, many artificial intelligence researchers have tried to tackle the problem of OCR complexity to build efficient OCR systems capable of working in accurate and real-time fashion (e.g., [2,30–33]). Although there are many OCR methods and toolsets available now in the literature, here we limit the work to a comparative study of four well-known OCR toolsets namely "Google Docs OCR", "Tesseract", "ABBYY FineReader", and "Transym".

Google Docs OCR [20] is an easy-to-use and highly available OCR service offered by Google within the Google Drive service [34]. We can convert different types of image data into editable text data using Google Drive. Once we upload an image or a PDF file to the Google Drive, we can start the OCR conversion by right-clicking on the file to select "Open with Google Docs" item, then the image is inside a Google Doc document and the extracted text is right below the image.

Tesseract was originally developed by HP as an open-source OCR toolset released under the Apache License [35], available for different operating system platforms, such as Mac OS X, Linux, and Windows. Since 2006, Tesseract developments have been maintained by Google [36], and it is among one of the top OCR systems used worldwide [29]. The Tesseract algorithm, at step one, uses adaptive thresholding strategies [37] to convert the input image into a binary one. It then utilizes connected component analysis to extract character layouts in which such layouts are then turned into *blobs*, the regions in an image data that differ in some part of the properties including color or intensity, compared to surrounding pixels [36]. *Blobs* are then formed as text lines, and consequently examined for an equivalent text size which is then divided into words using fuzzy spaces [36]. Text recognition will then proceed in a two stage process. In the first stage, the algorithm tries to discover each word from the text. Then, every satisfactory word will be passed to an adaptive classifier to train the data in stage one. In the second stage, the adaptive classifier assists to discover text data in a more reliable way [36].

ABBYY FineReader as an advanced OCR software system has been designed and developed by an international company, namely "ABBYY" [23] to provide high level OCR services. It has been improving the main functionalities of optical character recognition for many years, providing promising results in text retrieval from digital images [28]. The underlying algorithms of ABBYY FineReader have not yet been illustrated to the research community, probably because it is a commercial software product, and the package is not available as open-source code. Researchers and developers can access the ABBYY FineReader OCR by two different ways: (1) The ABBYY FineReader SDK which is available at https://www.abbyy.com/resp/promo/ocr-sdk/, and (2) Employing a web browser to try it over the Internet at https://finereaderonline.com/en-us/Tasks/Create.

Transym is another OCR software package that assists research and development communities in extracting accurate information from digital documents, particularly scanned and digital images. The source code of the Transym and its underlying algorithms are not available, but it has been delivered as a SDK which provides a high level API, and it also has a software package with a light GUI (graphical user interface) which can be easily installed and used efficiently. Transym OCR package along with some sample codes are available at http://www.transym.com/download.htm.

3 Applications in Healthcare Informatics

There have been limited studies surrounding the application of OCR within healthcare. Generally, the studies are divided into two major approaches: (1) Prospective data collection using forms that are specifically designed to capture hand printed data for OCR processing, and (2) Retrospective OCR data extraction using scanned historical paper documents or image forms [38]. There are several innovative examples of prospective OCR data capture at point-of-care. Titlestad [39] created a special OCR form to register new cancer patients into a large cancer registry. The OCR forms captured basic patient demographics and cancer codes. More recently OCR was introduced to capture data on anti-retroviral treatment, drug switches and tolerability for human immunodeficiency virus (HIV-1) patients [40]. This application enabled clinical staff to better manage the care of the HIV patient because the data could be tracked from visit to visit. Lee et al. [40] used OCR to minimize the transcription effort of radiologists when creating radiology reports. The Region of Interest (ROI) values (including area, mean, standard deviation, maximum and minimum) were limited to view on the computed tomography (CT) console or image analysis workstation. This image was then stored in a Picture Archiving and Communicating System (PACS). Radiologists would review the PAC images on the screen and then type the ROI measurements into a radiology report. OCR was used to automatically capture the ROI and measurements to place it on the clip board so it could be copied into the radiology report. Finally, Hawker et al. [41] used a set of cameras to capture the patient name when processing lab samples. OCR was used to interpret the patient name on incoming biological samples and then the name was compared to the laboratory information system for validity. The OCR mislabeling identification process outperformed the normal quality assurance process.

The majority of retrospective OCR studies have focused on retrieving medical data for research use. Peissig et al. [42] used OCR to extract cataract subtypes and severity from handwritten ophthalmology forms to enrich existing electronic health record data for a large genome-wide association study. This application extracted data from existing clinical forms that were not designed for OCR use with high accuracy rates. Fenz et al. [43] developed a pipeline that processed paper-based medical records using the open-source OCR engine Tesseract to extract synonyms and formal specifications of personal and medical data elements. The pipeline was applied on a large scale to health system documents and the output then used to identify representative research samples. Finally, OCR was applied to photographed printed medical records to detect diagnosis codes, medical tests and medications enabling the creation of structured personal health records. This study applied OCR to a real-world situation and addressed image quality problems and complex content by pre-processing and using multiple OCR engine synthesis [44].

4 Experimental Validations

To validate the accuracy, reliability, and performance of the Google Docs OCR, Tesseract, ABBYY FineReader, and Transym, several experiments on real, and also synthetic data were performed. In Sect. 4.1 we discuss the experimental setup, including the proposed dataset along with the testbed and its configurations. In Sect. 4.2 the qualitative OCR visualization results achieved from the OCR packages/services are reported. Subsequently, in Sect. 4.3 we examine the accuracy and reliability of the OCR systems, and perform a quantitative comparative study. Section 4.3 also presents and compare a set of quality attributes that the OCR systems offer to the research community.

4.1 Experimental Setup: The Dataset and Testbed

We have gathered 1227 images from 15 categories, including: (1) Digital Images, (2) Machine-written characters, (3) Machine-written digits, (4) Hand-written characters, (5) Hand-written digits, (6) Barcodes, (7) Black and white images, (8) Multi-oriented text strings, (9) Skewed images, (10) License plate numbers, (11) PDF files including electronic forms, (12) Digital receipts, (13) Noisy images, (14) Blurred images, and (15) Multilingual text images. Figure 1 shows an example from every category listed here. Except the PDF files (dataset No. 11), all images were taken in different resolutions using multiple formats, such as JPEG, TIFF, PNG, etc. The dataset attributes are explained in Table 1. Every dataset came up with the ground truth information including a list of the characters existing in the images. For all experiments illustrated here, we used 64-bit MS Windows 8 operating system on a personal computer with 3.00 GHz Intel Dual core CPU, 4 MB cache and 6 GB RAM. To communicate with Google Docs OCR [20], we employed Mozilla Firefox Version 48.0.1 at https://www.mozilla.org/.

4.2 Qualitative OCR Visualization

Using different images from the dataset illustrated in Sect. 4.1 we examined the qualitative visualization of the OCR systems. Figure 2 shows some sample results in extracting text data from digital images.

4.3 Comparative Study

Here, we further analyzed and compared the accuracy and reliability of the Google Docs OCR, Tesseract, ABBYY FineReader, and Transym using the dataset reported in Table 1. A detailed comparative study is reported in Table 2.

Table 1. Dataset attributes. First column shows the image categories. Number and type of the images is shown in the second column. CG, BW, and BWC stands for color and gray-scale, black & white, and black & white and color images respectively.

Image category	Images	Formats
Digital images	131 CG	TIFF, JPEG, GIF, PNG
Machine-written characters	47 CG	TIFF, JPEG, GIF, PNG
Machine-written digits	28 CG	TIFF, JPEG, GIF, PNG
Hand-written characters	49 BW	TIFF, JPEG, GIF, PNG
Hand-written digits	28 BW	TIFF, JPEG, GIF, PNG
Barcodes	224 BWC	TIFF, JPEG, GIF, PNG
Black and white images	101 BW	TIFF, JPEG, PNG
Multi-oriented text string	27 CG	TIFF, JPEG, PNG
Skewed images	93 CG	JPEG, PNG
License plate numbers	204 CG	JPEG, PNG
PDF files	14 CG	PDF
Digital receipts	108 CG	JPEG, PNG
Noisy images	24 CG	JPEG, PNG
Blurred images	31 CG	JPEG, PNG
Multilingual text images	118 CG	TIFF, JPEG, PNG

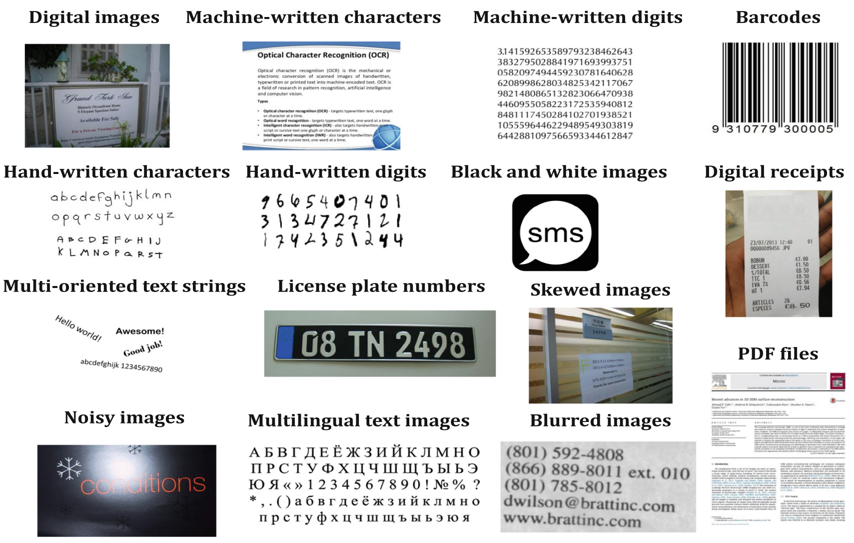


Fig. 1. Sample images from each category of the proposed dataset. The dataset includes 1227 digital images in 15 different categories.

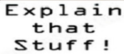
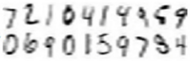
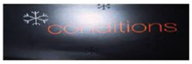
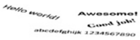
Image category	Sample image	Google Docs OCR	Tesseract	ABBYY FineReader	Transym
Digital images		RAE BUTLER BUILDING	RAB BUTLER BUILDING	Failed!	RAB BUTLER BUILDING
Machine-written characters		Explain that StUff	Explain that Stuff!	Explain that Stuff	Explain that Stuf f !
Hand-written digits		72104 9j8 DG90 I 597B4	7L/MI'1M'7 0(9401547'54	72104 I D 94 I 59 4	7Ll04 I Y Db90 I 597b4
License plate number		Failed!	W8 02 H 6886	IND HB 02 H 6886	WB02 W 6886
Barcodes		0.1234" 56789	01234"56789	01234 5b789	Failed!
Digital receipt		Total Oued 53.80 CREDIT CARD 53.80	Total Owed 53.8) CREDIT FARO 53.8)	Total Owed 53.80 CREDIT CARD 53.80	T...tal Uved 53. REDIT t-FRD 53
Skewed images		Failed!	Failed!	Failed!	Failed!
Noisy images		Failed!	Failed!	Failed!	Failed!
Blurred images		agency	Failed!	Failed!	ingen(y
Multi-oriented text		A/ S/o ho, Awesome! abcdefghijk 1234567890	YG//o Awesome! 3 3 abcdefghijk 1234567890	Awesome!	Awesome! Go abcdefghijk 1234567890

Fig. 2. The qualitative visualization of the four OCR systems using some sample images from the dataset.

A comparative examination of color as well as gray-scale images, with or without applying low-level image processing tasks (e.g., contrast/brightness enhancement) is shown in Fig. 3. To calculate the accuracy for every OCR systems discussed in the current work, we divided the number of characters which correctly extracted from a dataset by the number of characters existing in the same dataset using the Eq. (1), where n denotes the number of images in the dataset. Then, we calculated an average to obtain the total accuracy for each individual OCR system.

$$Accuracy = \frac{\sum_{k=1}^{n}(numberof\ correctly\ extracted\ characters)}{\sum_{k=1}^{n}(number\ of\ total\ characters\ in\ the\ dataset)} \times 100 \qquad (1)$$

Table 2 shows that the Google Docs OCR and ABBYY FineReader produced more promising results on the stated dataset, and the population standard deviation of accuracy obtained by those two are further consistent across the dataset. In addition to the experiments illustrated in Table 2, we divided the dataset into two parts including color and gray-scale images. Using color images, we obtained 74%, 64%, 71%, and 59% accuracy for the Google Docs OCR, Tesseract, ABBYY FineReader, and Transym respectively. After performing low-level image processing tasks including brightness and contrast enhancements, we obtained 75%,

Table 2. A Comparative study of the OCR systems. In this table we report analysis results obtained from 15 different image categories, examining the ability of the OCR systems to correctly extract characters from images. The percentage in the table means accuracy (Eq. 1).

		Extracted characters			
Image category	Existing characters	Google Docs OCR	Tesseract	ABBY FineReader	Transym
Digital images	1834	1613 (87.95%)	1539 (83.91%)	1528 (83.31%)	1463 (79.77%)
Machine-written characters	703	569 (80.94%)	549 (78.09%)	574 (81.65%)	554 (78.81%)
Machine-written digits	211	191 (90.52%)	193 (91.47%)	193 (91.47%)	194 (91.94%)
Hand-written characters	2036	1254 (61.59%)	984 (48.33%)	1204 (59.14%)	960 (47.15%)
Hand-written digits	43	29 (67.44%)	11 (25.58%)	25 (58.14%)	10 (23.26%)
Barcodes	867	841 (97%)	844 (97.35%)	832 (95.96%)	845 (97.47%)
Black and white images	71	69 (97.19%)	69 (97.19%)	65 (91.55%)	61 (85.92%)
Multi-oriented text strings	106	68 (64.15%)	30 (28.3%)	75 (70.75%)	23 (21.7%)
Skewed images	96	38 (39.58%)	31 (32.3%)	36 (37.5%)	27 (28.13%)
License plate numbers	1953	1871 (95.8%)	1812 (92.78%)	1894 (96.98%)	1732 (88.68%)
PDF Files	15693	15409 (98.19%)	14121 (89.98%)	15376 (97.98%)	14133 (90%)
Digital receipts	3672	3256 (88.67%)	3341 (90.99%)	3302 (89.92%)	3077 (83.8%)
Noisy images	337	179 (53.12%)	161 (47.77%)	184 (54.6%)	169 (50.15%)
Blurred images	461	259 (56.18%)	263(57.05%)	282 (61.17%)	277 (60.09%)
Multilingual text images	3597	2831 (78.7%)	2474 (68.78%)	2799 (77.81%)	1740 (48.37%)
Standard Deviation		$\sigma = 18.19$	$\sigma = 25.56$	$\sigma = 18.02$	$\sigma = 25.79$

64%, 75%, and 62% accuracy (Fig. 3). Using gray-scale images, we obtained 77%, 71%, 78%, and 68% accuracy for the Google Docs OCR, Tesseract, ABBYY FineReader, and Transym respectively. After performing low-level image processing tasks, such as brightness and contrast enhancement, we achieved 81%, 72%, 79%, and 70% accuracy (Fig. 3).

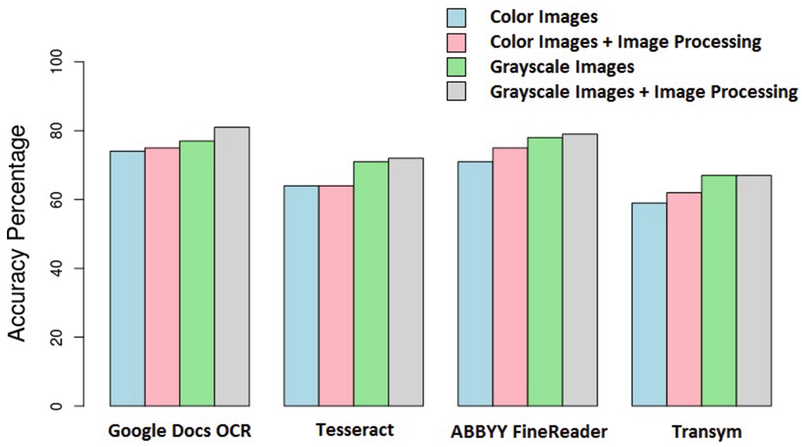


Fig. 3. A comparative study of the OCR systems using color and gray-scale images, with or without applying low-level image processing tasks (e.g., contrast/brightness enhancement). (Color figure online)

Table 3. A Comparative study of quality attributes of the OCR systems.

Quality attribute	Google Docs OCR	Tesseract	ABBYY FineReader	Transym
Open-source	No	Yes	No	No
Available online	Yes	No	Yes	No
Available as a SDK	No	Yes	Yes	Yes
Available as a Service	Yes	Could be	No	No
Multilingual support	Yes	Yes	Yes	Yes
Free	Yes	Yes	No	No
Operating systems	Any	Linux, Mac OS X, Windows	Linux, Mac OS X, Windows	Windows

Table 3 summarizes a comparative analysis of a set of quality attributes delivered by the OCR systems.

5 Discussion and Conclusion

We performed a qualitative and quantitative comparative study of four optical character recognition services, including Google Docs OCR, Tesseract, ABBYY FineReader, and Transym using a dataset containing 1227 images in 15 different categories. In addition to experimentally evaluating the OCR systems, we also reviewed OCR applications in the field of healthcare informatics. Based on our experimental evaluations using stated dataset, and without employing advanced image processing procedures (e.g., denoising, image registration), the Google

Docs OCR and ABBYY FineReader produced more promising results, and their population standard deviation of accuracy remained consistent across different types of images existing in the dataset. As we have seen in the experiments, the quality of input images has a crucial impact on the OCR outputs. For example, all of the examined OCR systems have faced a problem with skewed, blurred, and noisy images. The remedy can be sought in taking advanced low-level and medium-level image processing routines into account. We believe that the proposed dataset came with a reasonable distribution concerning the image types, but testing large-scale datasets employing hundred of thousand of digital images is still needed. As a classic machine learning problem, OCR is not only about character recognition itself, but also about learning how to be more accurate from the data of interest. The OCR is a challenging research topic that broadly lies in a variety of functionalities, such as layout analysis, support of different alphabets and digits style, in addition to well-formed binarisation to separate text data from an image background. As part of our future work, an attempt will be made to evaluate further OCR services using large-scale datasets, incorporating more significant statistical analysis for the accuracy and reliability. We will take advantage of advanced image processing algorithms and examine the benefit of their use towards developing more accurate and efficient optical character recognition systems.

Acknowledgement. The authors of the paper wish to thank Anne Nikolai at Marshfield Clinic Research Foundation for her valuable contributions in manuscript preparation. We also thank two anonymous reviewers for their useful comments on the manuscript.

References

1. Lin, H.-Y., Hsu, C.-Y.: Optical character recognition with fast training neural network. In: 2016 IEEE International Conference on Industrial Technology (ICIT), pp. 1458–1461. IEEE (2016)
2. Patil, V.V., Sanap, R.V., Kharate, R.B.: Optical character recognition using artificial neural network. Int. J. Eng. Res. Gen. Sci. **3**(1), 7 (2015)
3. Spitsyn, V.G., Bolotova, Y.A., Phan, N.H., Bui, T.T.T.: Using a haar wavelet transform, principal component analysis and neural networks for OCR in the presence of impulse noise. Comput. Opt. **40**(2), 249–257 (2016)
4. Bunke, H., Caelli, T.: Hidden Markov Models: Applications in Computer Vision, vol. 45. World Scientific, River Edge (2001)
5. Gupta, M.R., Jacobson, N.P., Garcia, E.K.: OCR binarization and image pre-processing for searching historical documents. Pattern Recogn. **40**(2), 389–397 (2007)
6. Jadhav, P., Kelkar, P., Patil, K., Thorat, S.: Smart traffic control system using image processing (2016)
7. Afli, H., Qiu, Z., Way, A., Sheridan, P.: Using SMT for OCR error correction of historical texts. In: Proceedings of LREC-2016, Portorož, Slovenia (2016, to appear)

8. Kolak, O., Byrne, W., Resnik, P.: A generative probabilistic OCR model for NLP applications. In: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, vol. 1, pp. 55–62. Association for Computational Linguistics (2003)
9. Kolak, O., Resnik, P.: OCR post-processing for low density languages. In: Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing, pp. 867–874. Association for Computational Linguistics (2005)
10. Deselaers, T., Müller, H., Clough, P., Ney, H., Lehmann, T.M.: The CLEF 2005 automatic medical image annotation task. Int. J. Comput. Vis. **74**(1), 51–58 (2007)
11. Kaggal, V.C., Elayavilli, R.K., Mehrabi, S., Joshua, J.P., Sohn, S., Wang, Y., Li, D., Rastegar, M.M., Murphy, S.P., Ross, J.L., et al.: Toward a learning health-care system-knowledge delivery at the point of care empowered by big data and NLP. Biomed. Inf. Insights **8**(Suppl1), 13 (2016)
12. Pomares-Quimbaya, A., Gonzalez, R.A., Quintero, S., Muñoz, O.M., Bohórquez, W.R., García, O.M., Londoño, D.: A review of existing applications and techniques for narrative text analysis in electronic medical records (2016)
13. Herbert, H.F.: The History of OCR, Optical Character Recognition. Recognition Technologies Users Association, Manchester Center (1982)
14. Tappert, C.C., Suen, C.Y., Wakahara, T.: The state of the art in online handwriting recognition. IEEE Trans. Pattern Anal. Mach. Intell. **12**(8), 787–808 (1990)
15. Assefi, M., Liu, G., Wittie, M.P., Izurieta, C.: An experimental evaluation of apple siri and google speech recognition. In: Proccedings of the 2015 ISCA SEDE (2015)
16. Assefi, M., Wittie, M., Knight, A.: Impact of network performance on cloud speech recognition. In: 2015 24th International Conference on Computer Communication and Networks (ICCCN), pp. 1–6. IEEE (2015)
17. Hatch, R.: SaaS Architecture, Adoption and Monetization of SaaS Projects using Best Practice Service Strategy, Service Design, Service Transition, Service Operation and Continual Service Improvement Processes. Emereo Pty Ltd., London (2008)
18. Tafti, A.P., Hassannia, H., Piziak, D., Yu, Z.: SeLibCV: a service library for computer vision researchers. In: Bebis, G., et al. (eds.) ISVC 2015. LNCS, vol. 9475, pp. 542–553. Springer, Heidelberg (2015). doi:10.1007/978-3-319-27863-6_50
19. Xiaolan, X., Wenjun, W., Wang, Y., Yuchuan, W.: Software crowdsourcing for developing software-as-a-service. Front. Comput. Sci. **9**(4), 554–565 (2015)
20. Google docs (2012). http://docs.google.com
21. Tesseract OCR (2016). https://github.com/tesseract-ocr
22. Tesseract.js, a pure javascript version of the tesseract OCR engine (2016). http://tesseract.projectnaptha.com/
23. Abbyy OCR (2016). https://www.abbyy.com/
24. Abbyy OCR online (2016). https://finereaderonline.com/en-us/Tasks/Create
25. Transym (2016). http://www.transym.com/
26. Online OCR (2016). http://www.onlineocr.net/
27. Free OCR (2016). http://www.free-ocr.com/
28. Mendelson, E.: Abbyy finereader 12 professional. Technical report, PC Magazine (2014)
29. Rice, S.V., Jenkins, F.R., Nartker, T.A.: The fourth annual test of OCR accuracy. Technical report, Technical Report 95 (1995)
30. Bautista, C.M., Dy, C.A., Mañalac, M.I., Orbe, R.A., Cordel, M.: Convolutional neural network for vehicle detection in low resolution traffic videos. In: 2016 IEEE Region 10 Symposium (TENSYMP), pp. 277–281. IEEE (2016)

31. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015)
32. Shah, P., Karamchandani, S., Nadkar, T., Gulechha, N., Koli, K., Lad, K.: OCR-based chassis-number recognition using artificial neural networks. In: 2009 IEEE International Conference on Vehicular Electronics and Safety (ICVES), pp. 31–34. IEEE (2009)
33. Ye, Q., Doermann, D.: Text detection and recognition in imagery: a survey. IEEE Trans. Pattern Anal. Mach. Intell. **37**(7), 1480–1500 (2015)
34. Google drive (2012). http://drive.google.com
35. Apache license, version 2.0 (2004). http://www.apache.org/licenses/LICENSE-2.0
36. Smith, R.: An overview of the tesseract OCR engine (2007)
37. Bradley, D., Roth, G.: Adaptive thresholding using the integral image. J. Graph. GPU Game Tools **12**(2), 13–21 (2007)
38. Rasmussen, L.V., Peissig, P.L., McCarty, C.A., Starren, J.: Development of an optical character recognition pipeline for handwritten form fields from an electronic health record. J. Am. Med. Inf. Assoc. **19**(e1), e90–e95 (2012)
39. Titlestad, G.: Use of document image processing in cancer registration: how and why? Medinfo. MEDINFO **8**, 462 (1994)
40. Bussmann, H., Wester, C.W., Ndwapi, N., Vanderwarker, C., Gaolathe, T., Tirelo, G., Avalos, A., Moffat, H., Marlink, R.G.: Hybrid data capture for monitoring patients on highly active antiretroviral therapy (haart) in urban Botswana. Bull. World Health Org. **84**(2), 127–131 (2006)
41. Hawker, C.D., McCarthy, W., Cleveland, D., Messinger, B.L.: Invention and validation of an automated camera system that uses optical character recognition to identify patient name mislabeled samples. Clin. Chem. **60**(3), 463–470 (2014)
42. Peissig, P.L., Rasmussen, L.V., Berg, R.L., Linneman, J.G., McCarty, C.A., Waudby, C., Chen, L., Denny, J.C., Wilke, R.A., Pathak, J., et al.: Importance of multi-modal approaches to effectively identify cataract cases from electronic health records. J. Am. Med. Inform. Assoc. **19**(2), 225–234 (2012)
43. Fenz, S., Heurix, J., Neubauer, T.: Recognition and privacy preservation of paper-based health records. Stud. Health Technol. Inf. **180**, 751–755 (2012)
44. Li, X., Hu, G., Teng, X., Xie, G.: Building structured personal health records from photographs of printed medical records. In: AMIA Annual Symposium Proceedings, vol. 2015, p. 833. American Medical Informatics Association (2015)