International Journal of Engineering & Technology, 8 (1.11) (2019) 154-157

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

An open Architecture for enhancing performance of complex OCR applications

K. Jambi, H. Al-Barhamtoshy, A. Fattouh, A. Al-Ghamdi, F. Eassa, M. Khemakhem

Faculty of Computing and Information Technology
King Abdulaziz University
Jeddah, Saudi Arabia
**Corresponding author E-mail:{kjambi, hassanin, afattouh, abdmalaise, feassa}@kau.edu.sa*

Abstract

Taking advantages of the different existing computing environments, infrastructures and resources for running in the optimal and/or in a customized manner any given complex Optical Character Recognition (OCR) software or application constitutes nowadays a challenge. Indeed, we mean by a complex OCR software or application any document digitization and computerization process which includes either plenty heterogeneous documents to process as input and/or several strong and sometimes complex OCR techniques to use in order to achieve good accuracy (recognition) rates. Moreover, the diversity and the very high computing and storage powers provided by such computing environments and infrastructures in one hand and the lack of powerful software and tools allowing their optimal or good utilization in the other hand make the problem a challenge.

Consequently, this paper proposes a novel open architecture, which attempts to use properly such environments and infrastructures in order to run at least in a pseudo optimal and/or a customized manner any given complex OCR software or application. Actually, the two most important features, which make the proposed architecture original, are first, its complete independency from the different existing distributed infrastructures that can run any given complex OCR application. Second, its flexibility, which allows any new distributed infrastructure to be considered during the scheduling process of any given complex OCR application since the scheduler, is able to detect automatically and consider any added infrastructure. Our architecture presents several advantages, indeed, it improves drastically the performances of any given complex OCR application, it is platform and software independent in addition to its flexibility as described and explained later.

Keywords— *Open Architecture, Complex OCR applications, performance.*

1. Introduction

Actually, intensive experiments achieved on the digitization of several kinds of documents have shown that the only way to reach very good or acceptable recognition rate is using the most suitable OCR technique for each piece of text to recognize [1]. This means that we need first to segment any input text to its homogeneous sub -texts where each of which will be assigned to the most suitable OCR during the digitization process [1]. In addition and for some types of writing (mostly handwritten
ones) [2], we need to use some very strong complementary approaches in order to reach acceptable recognition rates.
In this case and due to the complexity of most of powerful OCR approaches and technique, the digitization process will be very slow if one uses conventional computers for such a purpose [3-17]. Consequently, we propose in this paper an open framework that can speed -up substantially and in a customized manner such processes. The diversity and the impressive capabilities of the existing distributed and parallel architectures and infrastructures in the one hand and the complexity of the tackled problem in the other hand motivated us in proposing this work. Indeed, powerful OCR approaches and techniques are mostly complex and require sometimes enough computing power and storage especially in case of dealing with plenty heterogeneous documents to computerize. Moreover, the existing parallel and distributed architectures and infrastructures provide enough computing and storage powers which are unfortunately not well exploited and used especially for complex OCR applications. In fact, we found few works in the literature which have focused on such a tackled problem [4-17]; this can be explained may be by the preoccupation of researchers community by the design of new OCR approaches and techniques which attempt to improve the recognition rate for some specific writings without focusing on the corresponding performances and the way to take advantages of the existing approaches and techniques.
This paper is organized as follows; an overview on the related works is provided in Section 2; the description and features details of the proposed architecture are given in Section

3. Performance evaluation with a case study will be detailed in the fourth section, and section 5 presents the conclusion and the future work.

2. Related Works

Actually and to the best of our knowledge, there is no previous work, which attempted to propose such an idea.

Next section, we will detail the proposed architecture, which will ease the utilization of the existing parallel and distributed infrastructures in order to speed up drastically any complex OCR process.

Actually, the literature is very rich in terms of proposed of the OCR approaches.

Unfortunately, we think that all of them have not been used in the right way. Of course, we mean by the right way, the way to use them according to the specific writing for which they are powerful. That is why; we propose this framework, which will be detailed in the next section.

collaborator based about each task in the job to be sent to the user.

3. Proposed Architecture

The proposed open architecture portrayed in Figure 1 is used to manage the execution of complex OCR applications on different platforms for enhancing their performance. The tasks will be executed in parallel for enhancing the job performance.

Resources collectors include many software modules for collecting the available resources in each platform. There is a module for each platform because the module calls the API function of the target operating system for collecting the resources of the target machine to be stored in the metadata of resources. Each sub - module collects the metadata of available resources and sends them to metadata manipulator module. The structures (schemas) of resources metadata are as follows:

Environment (environment-id, environment-name).

Example for environment name: cloud system, traditional servers, or datacenters or others.

Machine (machine-id, machine-name, operating system, environment-id)

CPU (processor-type, speed, no-of-cores, status, machine-id)

GPU (processor-type, speed, no-of-cores, status, machine-id)

Memory (type, size, status, machine-id)

Resources selector selects the most appropriate (free or low load) resources from the resources metadata and sends them to the scheduler module.

Scheduler module uses the best-fit algorithm for scheduling the received tasks. It receives the selected resources from the resources selector, chooses the most efficient and best resources for the received task, and produces ready tasks.

The allocator receives the ready tasks and sends each task to the selected resources by calling APIs of the target operating system. The collaborator module to be returned to the user collects the returned results of all tasks. The collaborator also calls the API functions of the target operating systems for collecting the results. The results are integrated by the

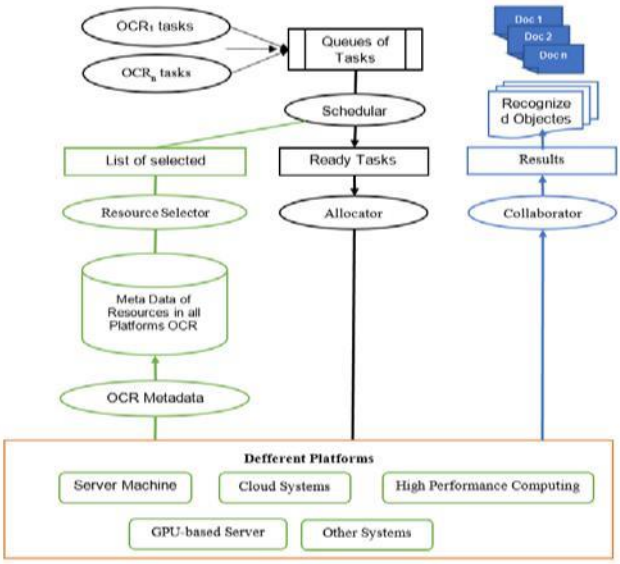

Fig. 1. The Proposed Architecture

4. Performance Evaluation

It is obvious that the proposed architecture enhances substantially the running time of any complex OCR applications since any given job, which is composed of n tasks, will be executed in a parallel manner.

Assume that the execution time of a:

$task_i = t_i$ where i=1,2,……n

If these tasks will be executed in a sequential manner then the
But if these n tasks will $\sum$ =1
completion time of the job will be the following:

Completion time (job) =
be executed in parallel then Completion time (job) = max (ti) where i=1, 2, …n

Case Study

Here, in this example, the input image is a multilingual text as shown in Fig. 1.

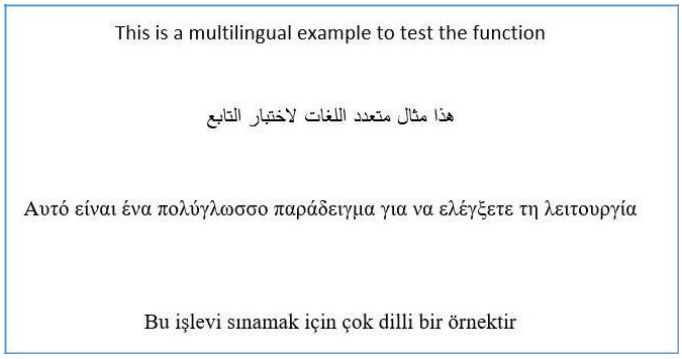

Fig. 2. A multilingual input image

The image has been segmented into homogeneous objects and four tasks are created as shown in Table 1.

Table 1 Generated Tasks

S	OCR	Input Image
1	English	

2	Arabic	This is a multilingual example to test the function هذا مثال متعدد اللغات لاختبار التابع
3	Greek	Αυτό είναι ένα πολύγλωσσο παράδειγμα για να ελέγξετε τη λειτουργία
4	Turkish	Bu işlevi sınamak için çok dilli bir örnektir

Accordingly, four workers are created to execute the four tasks. The tasks are first all submitted to workers to be executed serially, the execution result is shown in Figure 2. Then, each task is assigned to a worker and the execution result is shown in Figure 3.

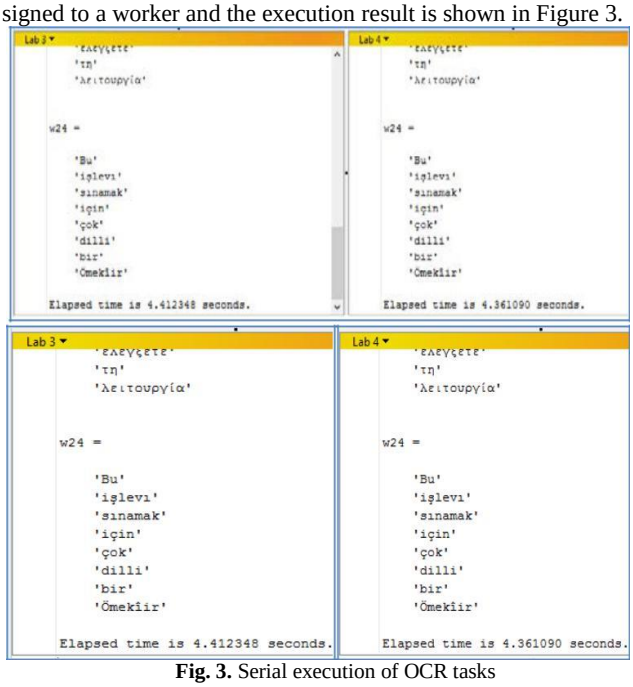

Fig. 3. Serial execution of OCR tasks

The minimum execution time in the serial case is 4.36 seconds, while the sum of execution times of four workers in the parallel case is 2.92 seconds, which means that assigning tasks to different workers reduces the execution time by 67% in this study case. This confirms the viability of our proposed work.

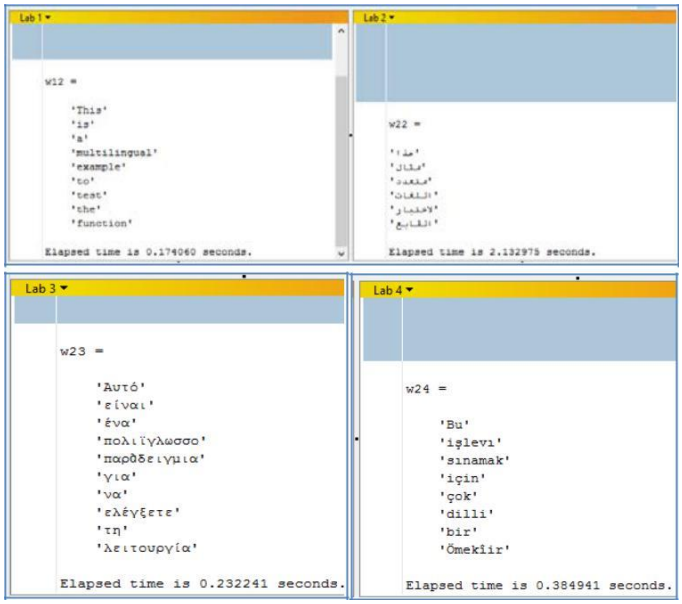

Fig. 4. Parallel execution of OCR tasks

5. Conclusion and Future Works

In this paper, we proposed a novel open architecture, which is, intended to speedup drastically any complex OCR process. The proposed architecture takes advantages of all the existing parallel and distributed infrastructures in order to reach the expected objective. The novelty or the added value of our proposal is first the corresponding flexibility since one could add any parallel or distributed infrastructure to the existing ones just by adding the corresponding metadata. Second, OCR developers or users can use it easily at will and can customize in the future the speedup of their complex OCR processes. Currently, we are studying the way to build the proposed architecture and intending its assessment and improvement in order to make it accessible by the research community.

References

[1] H. El Abed, L. Wenyin and V. Margner, International Conference on Document Analysis and Recognition (ICDAR 2011) Competitions Overview, International Conference on Document Analysis and Recognition, 2011.

[2] I. Abdelaziz, S. Abdou, and H. Al-Barhamtoshy, A large vocabulary system for Arabic online handwriting recognition, Pattern Analysis & Applications, Springer, Dec. 2015, DOI 10.1007/s10044-015-0526-7. http://link.springer.com/article/10.1007%2Fs10044-015-0526-7#page-1

[3] A. Hesham, S. Abdou, A. Badr, M. Rashwan, H. Al-Barhamtoshy, A Zone Classification Approach for Arabic Documents using Hybrid Features, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 7, No. 7, 2016, pp. 158-162, https://www.researchgate.net/profile/Sherif_Abdou/publication/305 82077 3_A_Zone_Classification_Approach_for_Arabic_Documents_usin g_Hyb rid_Features/links/57b1c50d08ae15c76cbb2e8b.pdf

[4] S. Eskenazi, P. Gomez-Krämer, J. Ogier, A comprehensive survey of mostly textual document segmentation algorithms since 2008. Pattern Recognition 64 (2017) 1–14.

[5] D. Petcu, S. Panica, D. Banciu, V. Negru, A. Eckstein, Optical Character Recognition on a Grid Infrastructure», 3rd International Conference on Automated Production of Cross Media Content for Multi-channel Distribution, IEEE, Nov, 2007.

[6] M. Khemakhem, A. Belghith: Agent based architecture for Parallel and Distributed Complex Information Processing, the International Revue on computers and software (IRECOS), Vol.2, No.1, p. 38-44, January 2007.

[7] M. Khemakhem, A. Belghith and M. Labidi: The DTW data distribution over a grid computing architecture, International Journal of Computer Sciences and Engineering Systems (IJCSES), Vol.1, No. 4, p. 241-247, December 2007.

[8] M. Labidi, M. Khemakhem and M. Jemni, Grid'5000 Based Large Scale OCR Using the DTW Algorithm: Case of the Arabic Cursive Writing, in the book, Recent Advances in Document Recognition and Understanding, ISBN 978-953-307-320-0 InTech, Rijeka, Croatia, October, 2011.

[9] Z. Trifa, M. Labidi and M. Khemakhem: Arabic Cursive Characters Distributed Recognition using the DTW Algorithm on BOINC: Performance Analysis, the International Journal of Advanced Computer Science and Applications, Vol. 2 No. 3, March 2011.

[10] H. Hamdi and M. Khemakhem: Distributing Arabic Handwriting Recognition system based on the combination of Grid Meta-Scheduling and Peer-to-Peer Technologies (Omnivore), the Universal Journal of Computer Science and Engineering Technology, 1(1), 31 – 35, Oct, 2010.

[11] M. Khemakhem, A. Belghith: Towards a distributed Arabic OCR based on the DTW algorithm, the International Arab Journal of Information Technology (IAJIT), Vol. 6, No. 2, p. 153-161, April 2009.

[12] M. Khemakhem, A. Belghith: A P2p Grid Architecture for Distributed Arabic OCR Based On the DTW Algorithm », The International Journal of Computers and Applications (ACTA PRESS, IJCA), Vol. 31, N°. 1, 2009.

[13] H. Hamdi, Kay Dornemann and M. Khemakhem: Advanced Distributed Architecture for a Complex and Large Scale Arabic Handwriting Recognition Framework, Accepted paper in the International Journal of High Performance Computing and Networking, IJHPCN, 2016.

[14] H. Hamdi, M. Khemakhem and Aisha Zaidan: Complementary Approaches built as Web Services for Arabic Handwriting OCR Systems via Amazon Elastic MapReduce (EMR) Model, Accepted pa-

per in the International Arab journal of Information Technology IAJIT 2016.
[15] H. Hamdi and M. Khemakhem: A Secured Distributed OCR System in a Pervasive Environment with Authentication as a Service in the Cloud, in Proc of the (IEEE) International Conference on Multimedia Computing and Systems, April 14-16, 2014, Marrakesh, Morocco.
[16] H. Hamdi and M. Khemakhem: Arabic Islamic Manuscripts Digitization based on Hybrid K-NN/ SVM Approach and Cloud Computing Technologies, Accepted paper in Taibah University International Conference on Advances in Information Technology for the Holy Quran and Its Sciences (NOORIC2013), December 2013, Al-Madinah, Saudi Arabia.
[17] H. Al-Barhamtoshy, M. Khemakhem, F. Eassa, A. Fattouh, A. Al-Ghamdi, K. Jambi, Universal Metadata Repository for Document Analysis and Recognition, The 13th ACS/IEEE International Conference on Computer Systems and Applications, AICCSA 2016.