International Journal of Computer Science, Communication & Information Technology (CSCIT)
Vol.2-Issue.2 - 2015
pp. 1-15

Pattern Recognition System based on Distributed Computing Architectures: Clusters, Peer to Peer and Data grid.

Hassen Hamdi[1,2], Khemakhem Maher[2,3]

[1] Computer science departement, College Of Science And arts at Al Ola, Taibah University, KSA
[2]Mir@cl Lab, FSEGS University of Sfax, Tunisia
[3] Computer Science Department, Faculty of Computing and Technology, University of King Abulaziz, KSA

{ hhassen2006@yahoo.fr ; maher.khemakhem@fsegs.rnu.tn }

Abstract.
Arabic Cursive handwriting recognition is challenging for many important and real-world applications such as document authentication, form processing, postal address identification, bank check recognition, manuscripts recognition , interpretation of historical documents and Islamic manuscripts. Therefore, in the last few decades, researchers and research centers have put an enormous effort into developing various and robust techniques but limited for handwriting recognition of small and medium documents. This chapter reviews existing handwriting recognition techniques and the current state of the art in cursive handwriting recognition and presents various hardware solutions for large scale Arabic handwritten character recognition system . This work deals with offline distributed handwriting recognition system based on distributed computing architecture. We present a pattern recognition system for large amount of document for isolated handwritten Arabic words.

Our recognition system is a Distributed Optical Character Recognition (DOCR) application via distributed computing architecture. The originality of our approach is the way we deal with large amount of Arabic Manuscript to digitize . We have introduced a new Arabic Handwriting Pattern Recognition designed to take advantage of distributed computing architecture. We have demonstrated that our approach present a very interesting framework which can lead to the implementation of powerful and speedup handwriting Pattern Recognition Systems.

The experiments were conducted on the Omnivore platform: Grid computing Meta-Scheduling system and P2P Technologies in the Department of Mathematics and Computer Science, University of Marburg, Germany, with a real large scaled dataset from the IFN/ENIT database.

Experimental results prove the validity of our approach to speed up the pattern recognition process.

Keywords: Arabic handwriting, Pattern Recognition, Distributed Computing, Omnivore, scalable.

1 Introduction

In many national libraries and archive centers, most of documents are still in their initial form and not yet digitized. These documents are indeed very rich in knowledge but leaving these precious documents in such a status constitutes enormous risks such as losing them and depriving especially researchers and students around the world of the corresponding knowledge and expertise.

The digitization of this documents focuses on the preservation of, and provision of access to, old manuscripts and other rare historical documents.

Important movement has been made in the Arabic handwriting Pattern Recognition System over the last few years. Most Arabic handwriting recognition application considered small and medium documents (few documents). Pattern recognition system that treat large amount of document needs enough power computing and storage capacities.

In order to solve both processing and storage capacities problem for the digitization of large amount of Manuscripts, we have opted for distributed computing architecture, more specifically, clusters, Peer to Peer and Data Grid paradigm that promise to eliminate the need for maintaining enormous computing facilities by companies and institutes.

Typically, many workstations in the faculties and student computer pools have a low utilization, especially during nights and weekends. Thus , the compute cluster at our university is used to perform the computations.

The remaining part of this chapter is organized as follows. First, it begins with a presentation of the concepts of pattern recognition. Section 2 reviews existing handwriting recognition techniques and also presents the current state of the art in cursive handwriting recognition. Section three describes the problem statement. Our approach is presented in the forth section. Section 5 provides some performance evaluation and investigation of our approach. Finally, our work is concluded by concluding remarks and future work (Section6).

2 Pattern recognition system

Pattern Recognition (PR) is the scientific fields that consist to classify objects into a number of categories or classes. Optical Character Recognition (OCR) is a sub-field of PR. This discipline is the mechanical or the electronic translation of scanned images of handwritten, typewritten or printed text into machine-encoded text. It is widely used to convert books and documents into electronic files, to computerize a record-keeping system in an office, or to publish the text on a website. The process of the OCR system is based on two steps learning and recognition. Each step can be broadly broken down into five stages: Pre-processing, segmentation, feature extraction, classification and post-processing. Figure below presents the OCR system .

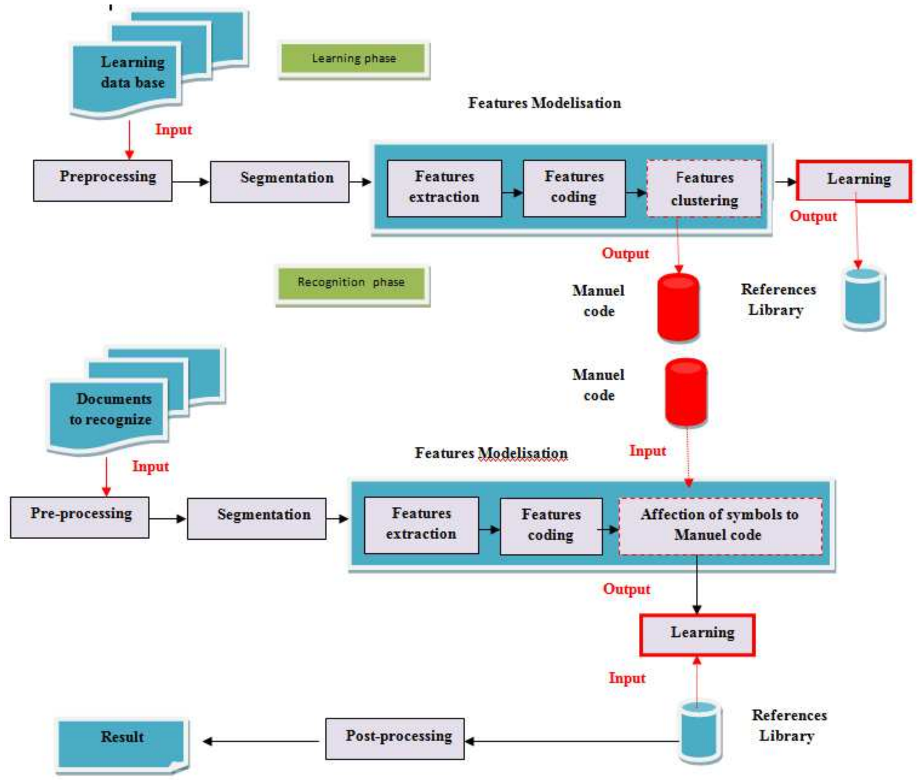

Fig. 1. OCR system

Pre-processing is the preliminary step which transforms the data into a format that will be more easily and effectively processed. This stage aims to produce data that are easy for the OCR systems to operate accurately. Pre-processing main objectives are: Deskewing, scaling, noise Elimination, slant Estimation and Correction, contour smoothing, thinning[1].

The Deskewing process is the first detecting step after the handwritten word has been written in on line OCR system or acquisition in off line OCR system. This task consists on rotating the word if the slope's angle is too high so the baseline of the word is horizontal. Many techniques for Deskewing task are presented in [2] [3] .

The Scaling preprocessing step sometimes may be necessary to produce words of relative and precise size. In the case of Burges, Be, and Nohl (1992), Different techniques are used such as the neural network . This technique accepted areas between the upper and lower baselines of each word as input. But, it was important to scale the words so that all cores were of an identical and unique height [4].

The noise may be introduced easily into an image during image acquisition. This nose can be small dots or blobs. Hence noise elimination or reduction in word images is necessary for further processing tasks. Morphological opening operation is one of different techniques used to remove noise in cursive handwritten words [5].

The slant estimation and correction task is an important part of any preprocessing step. Different techniques and algorithm are used to accomplish this task. Bozinovic and Srihari in [6] employed an algorithm consists on estimating the slant of a word by isolating those parts of the image that represented near vertical lines. then, an average estimation of the slant given by the near vertical lines was produced. The word was then slant corrected by applying transformation techniques such as Fourier transform. In this technique, the correction procedure of slants was necessary for segmenting their words using vertical dissection methods. Other estimation and correction techniques have been presented in the literature. Most scientists have accomplished this task using the chain code histogram techniques of the entire border pixels [7] [8] while other scientists have used vertical projections at various angles to analysis results [9].

Segmentation is a major and critical stage in the OCR system that seeks to decompose a word image of a sequence of characters into sub images of individual characters. This step incorporates text segmentation, line segmentation and character segmentation. Many techniques are used to accomplish this mission such as Hough Transform, horizontal projections, vertical projections, smearing and word extraction, connected component analysis [10].

Research surveys on segmentation confirmed that segmentation is one of the most difficult processes in cursive handwriting recognition process[11][12]. Some other work has produced encouraging and important results for the segmentation of cursive handwriting [13] [14][15].

Feature extraction can be defined as the process of extracting different features from the matrices of digitized characters. The objective of the feature extraction stage is to represent each character by an invariant feature vector which eases and maximizes the recognition rate with the least amount of data. A number of features have been found in literature on the basis of the OCR system. Features of a character can be classified statistical, structural and global transforms and moments [16]. The major problem in feature extraction step and especially of cursive/segmented character recognition is through the use of a variety of feature extraction algorithms and techniques. But, the extraction of appropriate features has proved difficult based on three important problems. First, the ambiguity of characters without the context of the entire word, second, the illegibility of certain characters due to the nature of cursive writing and finally difficulties in character classification due to anomalies and problems introduced during the segmentation step[17] [18].

Classification refers to one of the following tasks: first, classification of characters; second, classification of words and finally classification of features. In this stage is usually done by comparing the feature vectors corresponding to the input character with the representative of each character class. In this step, there is no such thing as the "best classifier". The use of classifier depends on many factors, such as an available training set, a number of free parameters etc. A number of classification methods were purposed by different researchers some of these are statistical methods[19], syntactic methods, template matching, Artificial Neural Networks(ANN), kernel methods such as k-Nearest neighbors (k-NN)[18], Bayes Classifier, Hidden Markov Models (HMM) [20], Support Vector Machines (SVM)[19], Euclidean distance[21]. It has also been prove that the use of multistage and hybrid classifiers has been very successful for classification step[22].

The post-processing stage is a series of techniques developed to post process noisy, multi font, no formatted OCR data on a word. The post-processing aims to determine if a field is alphabetic or numeric, verify that an alphabetic word is legitimate, fetch from a dictionary a set of potential entries using a garbled word as a key. The main objective of this stage is to improve the recognition rate by refining the decisions taken by the previous stage[23].

3 Existing handwriting recognition systems and techniques

An enormous number of conference and journal papers have been published in the handwriting recognition literature. The table below presents some performances, databases used, techniques and algorithms of exiting handwriting recognition system.

Table 1. some Arabic Handwriting Recognition System (AHRS)

Authors	Recognition Rate (%)	Techniques and algorithms	Database used
Mezghani et al. [24]	83.43	KNN	The training set contained 5,000 samples and the testing set about 2,400 samples of Arabic letters
Mezghani et al. [25]	93.54	Features vector based on Fourier descriptors + tangents vectors organization map, SOM	24,000 samples of Arabic letters for testing and 5,000 Arabic letters for training
Günter and al [26]	71.58	HMMs and Classifier Ensembles	IAM
Kherallah et al. [27]	90	Beta-elliptical model + SOM	24,000 Arabic digits
Jouini et al. [28]	90	Visual coding + NN	35,000 words for training 15,000 words for test
Halavati et al. [29]	95	Segmentation-based ap-	

		proach + template matching + fuzzy logic	
Biadsy et al. [30]	96.46	HMM	800 words * 4 times for training
Baghshah et al. [31]	88	Fuzzy approach using FLVQ algorithm	Isolated persian characters written by 128 persons
Elanwar et al. [32]	74	Geometric features based on Feeman chain + segmentation- based approach using dynamic programming and template matching	317 words (1,814 characters), written by four writers for training
Kherallah et al. [33]	95.08	Modeling based on inflection point detection, the overlapped form of beta signals, and the elliptic arcs + beta-elliptical modeling + combining MLPNN + SOM + FKNN	30,000 Arabic digits
Izadi et al. [34]	89.4 % for two letters word 85 % for three letters word	Wavelet-based smoothing technique + Segmentationbased approach + DTW classifier	20 classes of paws with two and three characters for Persian script
Daifallah et al. [35]	85.3 - 92.6% forwords 88.8–97.2 % for letters	Segmentation approach + HMM letters without marks or point	150 words , 720 letters inside words
Ghods and Kabir [36]	94	Geometric features + decision tree + minimum distance classifier	4,000 isolated letters from “TMU dataset” written by 117 writers
Biadsy et al. [37]	98.44	Geometric features + holistic approach for word-part recognition using HMM + word-part dictionary and the letter-shape models	3,200 words for training 2,358 words for test with 10 writers
Eraqi and Abdelazeem [38]	87 %	Grapheme segmentation + offline features + Fuzzy SVM	ADAB database

As a conclusion , in the last few decades, researchers and researches center have put an enormous effort into developing various and robust techniques but limited for handwriting recognition of small and medium documents.

4 Problem statement

There are different Arabic teaching, instructors , research centers, researchers departments, but a little digital information and data is available about their activities and contributions and details of their expertise and wisdom are not well known.

In many national libraries, there are several publications in the form of books, journals, research papers, conference proceedings, dissertations, and monographs. But, the number of comprehensive documentation centre is limited such as in Australia [39], in Tunisia , the National cultural heritage digitization project "RAED", conceived of by the Culture and Vocational Training and Employment Ministries [40], Algeria [41]. Hence there is an urgent need to develop a system for monitoring and facilitate the creation of digital library.

To ease the use of such documents, archive them and make them readable by a bigger audience, it is necessary to have them digitalized.

We assume that the documents will be there as scanned pages in shape of images. At the beginning, it comes to mind to use a unique computer to digitize the documents. Therefore, we started the digitalization of some documents as a sequence of words. In this case, different Arabic words are recognized sequentially on a PC (3.4 GHZ CPU frequency, 1GB of RAM and running Windows XP-professional). The time of recognition process achieved 5.85 minutes with a single document of 10000 words. Table 1 presents the results

Table 2. Variation of the speedup according to the size of the document

Size of the document	1000	2000	3000	4000	5000
Speedup(s)	1.15	1.27	1.50	2.07	2.40
Size of the document	6000	7000	8000	9000	10000
Speedup(s)	3.75	4.01	4.45	4.95	5.85

Results showed that it is obvious that this solution is not adaptable to a large and huge amount of documents.

Another motivation for the work presented in this paper originates with the existing of many workstations in the faculties and student computer pools that have a low utilization rate, especially at nights or weekends and holidays.

To sum up, the essential objective of our work is to integrate unused resources such us desktop computers into a Grid of cluster resources to speed up the OCR process.

5 The proposed approach

Handwriting recognition and especially Arabic handwriting recognition still constitutes a big challenge, especially if we need to digitize a big and large amount of documents and manuscripts, despite the wide range of proposed algorithms and techniques which attempted to solve the inherent problems [42]. The complex morphology and the cursive aspect of this writing are behind the weakness of the proposed approaches. Based on a survey of the existing proposed OCR systems lead to the conclusion that maybe the combination of some of them, which are very complementary, can lead to the development of an efficient Handwriting OCR systems. Unfortunately, such combination requires a huge amount of computing power owing the fact that most of these approaches and techniques are complex in terms of computing.

One of the most significant technologies to the Internet today is the distributed computing technologies such as Peer to Peer (P2P) and Grid computing.

Hopefully, theses distributed computing infrastructures can offer enough computing power and storage capacities which can be exploited and used to solve our problem. Hence, it is necessary to distribute the recognition system to speed up the used time and higher the throughput. This is possible because the recognition of a word can be seen as an atomic operation without any interconnection to the recognition of another word.

The Computing Grids paradigms are primarily used to connect different dedicated compute clusters. The building of dedicated compute clusters needs and requires a considerable administrative and fiscal resources. Often, necessary compute power is already available in the form of desktop computers - incorporating them into on-demand resource pools prevents investments in additional computer systems, alleviating the problem of resource wastage.

Peer To Peer system (P2P), is a network with any nodes can act both as a client or a server and pays its participation in the network by offering access to resources, most of peers processing power and/or disk space.

We propose a novel approach to distribute the Arabic handwriting OCR system based on the combination of Grid computing meta-scheduling system and Peer–to-Peer architectures, to integrate unused resource pools.

Our approach uses a Grid computing meta-scheduling system as an interface to the user. Using this front, the user experience no difference to standard Grid submission systems and only small differences to cluster computing scheduling systems. In our case, we use the GridWay interface as deployed Grid meta-scheduler system because it is widely used in Grid computing environments.

The GridWay technologies is a Service-oriented architecture based on many features such as flexibility and security. This architecture is a coordinated resource sharing infrastructure allowing dynamic service exchange and manage among members of different virtual communities. The Semantic and concept Grid computing highlights the information and knowledge area of these service exchanges [43].

In our approach, the P2P technologies are distributed systems characterized with some features such as auto-adaptive, self-healing, self-configuring and decentralized features. Our architecture is a distributed hash table (DHT) based P2P technologies. In these system all peers are equal. The P2P system complements the classic “Client/Server” model: each participant can be either Client or Server [10].

At this point it is important to establish the concept of jobs. When we talk about a job we are thinking of an executable combined with some data and described by a job description. There are some specifications as Job Submission Description Language (JSDL) used by GridWay that describes the submission aspects of a job or Resource Specification Langage (RSL) used within Grids and some proprietary as used by GridWay internally.

Omnivore [44], the Grid computing Meta-Scheduling system and P2P Technologies platform created in the Department of Mathematics and Computer Science, University of Marburg, Germany is used in our approach. this architecture is the interface between GridWay and our P2P meta-scheduler system. This means the P2P system can be either used to schedule between Grid sides without using another Grid meta-scheduler such as GridWay, but also as a meta-scheduler interfacing between a Grid meta-scheduler such as GridWay and Grid sides. At least it could be used as a classic scheduler scheduling between desktop computers, called just P2P scheduling. Using this technologies, Omnivore and the P2P scheduling system proposed by us is very flexible and easy. To achieve this goal, the system offers a plug-in interface. The P2P scheduler supports running jobs directly on desktop computers or in virtual machines on desktop computers. To ease the reading of our paper we subsumed Omnivore and the P2P scheduling system as Omnivore. This platform is essentially thought for incorporating unused desktop computers within a PC pool and teaming them up with Grid computing architecture .

Contrary to the GridWay platform that can be configured only with Linux Operating system , the Omnivore architecture can be configured with different others Operating system such as Windows and Mac. Hence it was important to extend the job description file with some modifications using JSDL. The new necessary information for Omnivore, is hidden within the existing environment parameter. This define the environment variables for submission of the job. Below, is an example of a GridWay job description file.

```
EXECUTABLE=/does/not/matter
ARGUMENTS=-jar OCR.jar test
ARGUMENTS=-la /tmp
ENVIROMENT=EXEC=LOCAL, LOCALBINARY=java.exe
EXECUTABLE=/bin/ls
ARGUMENTS=-la /tmp
```

In this GridWay job description file, the executable can be only be integrated as a Linux binary that it is not possible for Windows OS platform. Omnivore resolve this limit, it ignores the EXECUTABLE parameter and access to the real executable file from the hidden information LOCALBINARY.

Also, this information is used to choose environment where the job should be done. At the moment, the Omnivore platform supports many features: first, LOCAL when we want to execute jobs directly on the system. Second, GLOBUS when we want to submit the job to a running Globus Toolkit Grid environment) and third some Virtualization environments such as cloud computing. The execution environment is characterized by the parameter EXEC. Local execution is used in our paper.

The basics of the proposed approach is to use Omnivore platform to the pattern recognition system to execute the parallelized OCR jobs. To parallelize them it is necessary to create packages containing a part of the data (learning data base and documents to recognize), in our case images, a small data base and an executable combined with the job description template.

In our case, we propose to split optimally the binary image of a given Arabic Manuscripts text to be recognized into a set of binary sub images and then assign them among some computers interconnected to the GridWay. The Grid Computing is composed of different institutions heterogeneous computers interconnected via the LAN network. One of these computers is named the coordinator and all others are named workers. The coordinator is the responsible of the management of the data and the recognition process and the coordination among workers. the coordinator node is working as a web server. To launch on the grid a distributed Arabic recognition process, we should first log in to the coordinator, then ask it some information like the number of PCs, the power computing, the storage capacity and the OS of available nodes (workers).

6 The experimental study

An experimental study is conducted to improve the influence of Omnivore architecture first on the the performance of our OCR system and especially execution time and second on the efficiency of the utilization of resources of our university. We have used different corpus with different size (1000,...9000 words) randomly chosen from the IFN/ENIT [45] corpus data base formed of handwritten Tunisian town's names. Also we have considered a reference library composed of 345 characters representing approximately the totality of the Arabic alphabet different position (rotation and translation), written with different scripter.

To analyze our experiments, three factors are defined: the execution time, the speedup factor and finally the efficiency factor. First, The execution time of a given task represents the time spent by the system executing that task. This time includes the time spent executing run time or system services on its behalf. Second, the speedup factor is the ratio of the elapsed time using sequential mode with just one processor to the during time using the distributed platform and third the efficiency factor that define the ratio of the speedup factor to the number of computers or clusters incorporated in the work. First, the OCR application was configured , implemented and tested on a pool of clusters and second on the Omnivore architecture. Clusters and Omnivore are interfaced by Gridway to have the same output.

6.1 Distributed OCR system based on clusters architecture

We start our tests with executing our OCR application on Clusters architectures. All OCR jobs are executed on Compute nodes characterized with 16 GByte memory, 2xDualCore Opteron 2216 HE 2.4GHz, 250 GByte SATA HD, and the network speed was 1 Gbit/s.
Table 2 shows some typical results of execution time, speed up factor and efficiency factor using clusters architecture.

Table 3. Variation of the OCR system performances using clusters architecture with 20 works.

	Execution time	Speed up factor	Efficiency factor
6000	0.43	8.80	0.44
7000	0.43	9.33	0.47
8000	0.45	9.89	0.49
9000	0.48	10.31	0.53

This table shows in particular that:

- The speedup factor increases with the number of Compute nodes used and the efficiency factor increases with the size of the file to recognize. The efficiency factor is greater than 0.54 which means that the computing power of each dedicated compute node is used for more than 54%.
- If we use 20 compute nodes then the speedup factor reaches the value 10.76 which amounts to a recognition rate around 660 characters per second which is a very interesting recognition speed compared to the existing products[46][47].

6.2 Distributed OCR system based on Omnivore platform

To distribute our OCR application, we have used 20 dedicated homogeneous nodes characterized by the exact same configuration: 3.4 GHZ CPU frequency, 1GB of RAM and running Windows XP-professional operating system, taken from a PC pool at the University of Marburg, Germany in a grid network with 100 Mbit/s capacity.
The experimental results of the distributed pattern recognition performance using Omnivore platform are presented in the following table .

Table 4. Variation of the OCR system performances using Omnivore architecture with 20 works.

	Execution time	Speed up factor	Efficiency factor
6000	0.33	11.00	0.55
7000	0.34	12.01	0.60
8000	0.37	13,50	0.68
9000	0.39	15.60	0.78

Table 3 shows the advantages of using distributed architecture based on Omnivore on the already defined factors.

The speedup factor increases with the number of workers used and the efficiency factor increase with the size of the file to recognize. The efficiency factor is greater than 0.78 with a file of 9000 words which means that the computing power of each worker is used for more than 78 %.

If we use a distributed architecture based on Omnivore with 20 workers then the speedup factor reaches the value 15.60 which amounts to a recognition rate around 840 characters per second which is a very interesting recognition speed compared to the existing products [48] and the results using a dedicated cluster.

7 Conclusion

With the emergence of distributed computing as a paradigm in which scientific computing can be done exclusively on resources leased only when needed from big data canters. Therefore, in this paper, the main goal is to answer the question is the performance of distributed computing sufficient for large scale Pattern recognition system?

To this end, we proposed the integration of the distributed computing paradigm in the pattern recognition system using Grid Meta-Scheduling system and P2P Technologies (Omnivore) for the design of the Arabic distributed OCR system to speed up the recognition process.

Performance evaluation of the proposed approach confirms that Omnivore can provide an effective framework to speed up the recognition process and integrate strong complementary approaches that can lead to the implementation of powerful handwritten OCR systems.

In addition, our future work also includes expanding the distributed pattern recognition system based on distributed computing technologies such as cloud computing. Although this study is limited to data distribution, extensions to the distribution of the OCR application (we examined how to distribute the different stages of the OCR system such as pre-processing, segmentation, feature extraction between nodes of Omnivore) can be attempted.

8 References

[1] Vamvakas G., Gatos B., Pratikakis I., Stamatopoulos N., A. Roniotis and S.J. Perantonis, "Hybrid Off-Line OCR for Isolated Handwritten Greek Characters", The Fourth IASTED International Conference on Signal Processing, Pattern Recognition, and Applications (SPPRA 2007), ISBN: 978-0-88986-646-1, pp. 197-202, Innsbruck, Austria, February (2007).

[2] Senior, A.W. Off-line cursive handwriting recognition using recurrent neural networks [unpublished doctoral dissertation, Cambridge, England: University of Cambridge. 1994.

[3] Brown, M.K., & Ganapathy, S. Preprocessing techniques for cursive script word recognition. Pattern Recognition, 16(5), 447–458. (1983).

[4] Burges, C.J.C., Be, J.I., & Nohl, C.R. Recognition of handwritten cursive postal words using neural networks. Proceedings of the 5th USPS Advanced Technology Conference pp.117–124. (1992).

[5] Chen, M.-Y., Kundu, A., Zhou, J., & Srihari, S.N. , Off-line handwritten word recognition using hidden Markov model. Proceedings of the 5th USPS Advanced Technology Conference , pp. 563–579. (1992).

[6] Bozinovic, R.M., & Srihari, S.N. Off-line cursive script word recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 11(1), 68–83, (1989).

[7] Ding, Y., Kimura, F., Miyake, Y., & Shridhar, M. Evaluation and improvement of slant estimation for handwritten words. Proceeings of the 5th International Conference on Document Analysis and Recognition Bangalore, India: IEEE Computer Society Press. pp. 753–756, (1999).

[8] Kimura, F., Shridhar, M., & Chen, Z. Improvements of a lexicon directed algorithm for recognition of unconstrained handwritten words. Proceedings of the 2nd International Conference on Document Analysis and Recognition Tsukuba, Japan: IEEE Computer Society Press. pp. 18–22, (1993).

[9] Guillevic, D., & Suen, C.Y. Cursive script recognition: A sentence level recognition scheme. Proceedings of the 4th International Workshop on the Frontiers of Handwriting Recognition , pp. 216–223. (1994).

[10] Saha S., Subhadip Basu, Mita Nasipuri and Dipak Kr. Basu, A Hough Transform based Technique for Text Segmentation, Journal of computing, volume 2, issue 2, ISSN 2151-9617 / February, 2010.

[11] Casey, R.G., & Lecolinet, E. A survey of methods and strategies in character segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 18(7), 690–706. (1996).

[12] Dunn, C.E., & Wang, P.S.P. Character segmentation techniques for handwritten text—Asurvey. Proceedings of the 11th International Conference on Pattern Recognition , pp. 577–580, (1992).

[13] Casey, R.G., & Lecolinet, E. A survey of methods and strategies in character segmentation. IEEE Transactions on Pattern Analysis andMachine Intelligence, 18(7), 690–706. (1996).

[14] Lu, Y. Machine printed character segmentation An overview. Pattern Recognition, 28(1), 67–80. (1995).

[15] Xiao, X., & Leedham, G. Knowledge based cursive script segmentation. Pattern Recognition Letters, 21(10), 945–954. (2000).

[16] Verma R., Jahid A., A-Survey of Feature Extraction and Classification Techniques in OCR Systems, International Journal of Computer Applications & Information Technology Vol. I, Issue III, ISSN: 2278-7720, . 2012

[17] Blumenstein, M., Liu, X.Y., & Verma, B. A modified direction feature for cursive character recognition. Proceedings of the International Joint Conference on Neural NetworksBudapest, Hungary: IEEE Computer Society Press, pp. 2983–2987,. (2004).

[18] Blumenstein, M., & Verma, B. Analysis of segmentation performance on the CEDAR benchmark database. Proceedings of the Sixth International Conference on Document Analysis and Recognition Seattle: IEEE Computer Society Press. pp. 1142–1146, (2001).

[19] Liu, C.-L., & Fujisawa, H. Classification and learning for character recognition: Comparison of methods and remaining problems. Proceedings of the International Workshop on Neural Networks and Learning in Document Analysis and Recognition pp. 5–7. Seoul, Korea: IEEE Computer Society Press. 2005

[20] Günter, S., & Bunke, H. Off-line cursive handwriting recognition using multiple classifier systems—On the influence of vocabulary, ensemble, and training set size. Optics and Lasers in Engineering, 43(3-5), 437–454. (2005).

[21] Jing Li, Bao-Liang Lu An adaptive image Euclidean distance, Pattern Recognition, pp 349 -- 357, (2009)

[22] Camastra, F., & Vinciarelli, A. Combining neural gas and learning vector quantization for cursive character recognition. Neurocomputing, 51, 147–159. (2003).

[23] Youssef Bassil, Mohammad Alwani, Ocr Post-Processing Error Correction Algorithm Using Google online spelling suggestion , Journal of Emerging Trends in Computing and Information Sciences, ISSN 2079-8407, Vol. 3, No. 1, January 2012

[24] Mezghani, N., Mitiche, A., Cheriet, M.: On-line recognition of handwritten Arabic characters using aKohonen neural network. In: Proceedings of IWFHR'02, pp. 490–495. Niagara on theLake, Canada (2002)

[25] Mezghani, N., Mitiche, A., Cheriet, M.: Combination of pruned Kohonen maps for on-line Arabic characters recognition. In: Proceedings of ICDAR'03, pp. 900–905, Edinburgh (2003)

[26] Günter, S., & Bunke, H. Feature selection algorithms for the generation of multiple classier systems and their application to handwritten word recognition. Pattern Recognition Letters, 25(11), 1323–1336, (2004).

[27] Kherallah, M., Njah, S., Alimi, A.M., Derbel, N.: Recognition of on-line handwritten digits by neural networks using circular and Beta approaches. In: Proceedings of IEEE International Conference SMC'02,. Hammamet, Tunisia, pp. 26–30 (2002)

[28] Jouini, B., Kherallah, M., Alimi, M.A.: A new approach for online visual encoding and recognition of handwriting script by using neural network system. In: 6th International Conference on. Artificial Neural Nets and Genetic Algorithms, pp. 161–166. Springer, Vienna (2003)

[29] Halavati, R., Jamzad, M., Soleymani, M.: A novel approach to persian online hand writing recognition. Trans. Eng. Comput. Technol. 6, 232–236 (2005)

[30] Biadsy, F., El-Sana, J., Habash, N.: Online Arabic handwriting recognition using hidden Markov models. In: Proceedings of the Tenth International Workshop on Frontiers in Handwriting Recognition, pp. 85–90 (2006)

[31] Baghshah, M.S., Shouraki, S.B., Kasaei, S.: A novel fuzzy classifier using fuzzy LVQ to recognize online persian handwriting. In: 2nd IEEE Conference on Information & Communication Technologies (ICTTA) (2006)

[32] Elanwar, R.I., Rashwan, M.A., Mashali, S.A.: Simultaneous segmentation and recognition of Arabic characters in an unconstrained on-line cursive handwritten document. In: Proceedings of World Academy of Science, Engineering and Technology (WASET), International conference on Machine learning and Pattern Recognition MLPR2007, vol. 23, pp. 288–291, Germany (2007)

[33] Kherallah, M., Haddad, L., Alimi, A.M., Mitiche, A.: Online handwritten digit recognition based on trajectory and velocity modeling. Pattern Recogn. Lett. 29, 580–594 (2008)

[34] Izadi, S., Haji,M., Suen, C.Y.: A new segmentation algorithm for online handwritten word recognition in Persian script. pp. 1140–1142, ICHFR (2008)

[35] Daifallah, K., Zarka, N., Jamous, H.: Recognition-based segmentation algorithm for on-line Arabic handwriting. In: Proceedings of International Conference on Document Analysis and Recognition, ICDAR 2009, pp. 877–880. Barcelona, Spain, IEEE (2009)

[36] Ghods, V., Kabir, E.: Feature extraction for online Farsi characters. In: ICFHR, 2010 12th International Conference on Frontiers in Handwriting Recognition, pp. 477–482 (2010)

[37] Biadsy, F., Saabni, R., EL-Sana, J.: Segmentation-free online Arabic handwriting recognition. Int. J. Pattern Recognit. Artif. Intell. 25(7), 1009–1033 (2011)

[38] Eraqi, H., Abdelazeem, S.: An on-line Arabic handwriting recognition system based on a new on-line graphemes segmentation technique. In: Proceedings of ICDAR 2011, pp. 409–413 (2011).

[39] Holley R., How Good Can It Get? Analysing and Improving OCR Accuracy in Large Scale Historic Newspaper Digitisation Programs. Lib Magazine, , vol. 15 no 3/4 , (2009)

[40] http://www.emploi.gov.tn

[41] zoheir H., le rôle des technologies dans le stockage du manuscrit arabe, cybrarians journal, n° 14, septembre 2007.

[42] Sangsawad S. and C. Fung Using Content Based Image Retrieval Techniques for the Indexing and Retrieval of Thai Handwritten Documents, IEEE Xplore., vol 1, june 2010.

[43] I. Foster and Carl Kesselman, editors. The Grid: blueprint for a new computing infrastructure.Morgan Kaufmann, San Francisco, CA, USA, 82, 84, 87, 1999.

[44] Dabek F., Zhao B., Druschel P., Kubiatowicz J., and Stoica I.. Towards a Common API for Structured P2P Overlays. In F. Kaashoek and I. Stoica, editors, Revised Papers from the 2nd International Workshop on P2P Systems (IPTPS' 03), volume 2735 of Lecture Notes in Computer Science, pages 33–44, Berlin, Heidelberg,. Springer-Verlag. (2003).

[45] Heidt M., Dörnemann T., Dörnemann K., and Freisleben B.. Omnivore: Integration of Grid Meta-Scheduling and Peerto- Peer Technologies. In Proceedings of 8th International Symposium on Cluster Computing and the Grid (CCGrid 08), pages 316–323, (2008).

[46] Pechwitz M., S. Maddouri S., Mrgner V., Ellouze N., and Amiri H.. Ifn/enit - database of handwritten arabic words. In In Proc. of CIFED 2002, pages 129– 136, (2002).

[47] CiyaICR product, http://www.Ciyasoft.com, (2004)

[48] Khemakhem M. and Belghith A.. Towards A Distributed Arabic OCR Based on the DTW Algorithm: Performance Analysis The International Arab Journal of Information Technology, Vol. 6, No. 2, (2009).