Diyala Journal for Pure Science

Parallel Processing Based on Fragmentation of English Text Pictures to Speed Up Recognition of Characters on a Multi-Core Computer

Maisa'a Abid Ali* and Abdul Monem S. Rahma

Computer Science Department – University of Technology – Baghdad – Iraq

*110003@uotechnology.edu.iq

Received: 11 January 2019 **Accepted: 16 June 2019**

DOI: https://dx.doi.org/10.24237/djps.15.04.498B

Abstract

Script picture analyses use algorithms to turn images of text into readable computer data. The goal of script picture analyses is pattern recognition of script and graphics combinations from images for extraction of the data. This paper solves a long-existing problem in pattern recognition in script images regarding recognition of characters and words using a computer processor. The proposed system rely on main steps of execute, first step: is load English script of image file, second step: is partition image file to several image pages, third step: is divide the image into number of blocks, and final step: multi-block method can be used in each row of the script image with parallel processing using a multi-core computer (one processor P1, two processors P2, three processors P3, and four processors P4) with rapid processing for script images for pattern recognition. The outcome obtained with a multi-core computer is very fast for processing script images and is efficient and accurate. It can use the modified algorithm in multi-processor system that available now and not used all the capacity in processing in one processor as in old computers.

Keywords: English script image, Pattern recognition, Character recognition, Block method, multi-core processors.

Diyala Journal for Pure Science

Parallel Processing Based on Fragmentation of English Text Pictures to Speed Up Recognition of Characters on a Multi-Core Computer

Maisa'a Abid Ali and Abdul Monem S. Rahma

طريقة تقسيم صورة نص في اللغة الانكليزية بالاعتماد على زيادة سرعة تمييز الحروف في تعدد النواة بالحاسبة

ميساء عبد علي و عبد المنعم صالح رحمة

قسم علوم الحاسوب – الجامعة التكنولوجية – بغداد – العراق

الخلاصة

تحليل صورة نص باستخدام خوارزمية تحويل صورة النص الى قراءة بيانات الصورة. الهدف في تحليل صورة النصوص هو تمييز الانماط في مجموعة من النصوص والرسومات لأستخلاص البيانات. هذا البحث يحل مشكلة تمييز الانماط لمدة طويلة لنصوص الصور وخصوصا تمييز الحروف والكلمات باستخدام معالج الحاسبة. ان اقتراح النظام يعتمد على عدة خطوات في التنفيذ، الخطوة الاولى: هي تحميل صورة نص اللغة الانكليزية، الخطوة الثانية: هي تجزئة الصورة الى عدد من الصفحات، الخطوة الثالثة: هي تقسيم الصورة الى عدد من البلوكات: والخطوة الاخيرة: تم استخدم طريقة تعدد البلوك لكل سطر لصورة النص مع استخدام معالجات متوازية اي تعدد النواة للحاسبة (معالج واحدP1، معالجات اثنين P2، معالجات ثلاثة P3، معالجات اربعة P4) مع سرعة المعالجة لتمييز نمط صور النصوص. الحصول على نتائج تعدد نواة الحاسبة هو سريع جدا في معالجة صور النصوص و كفوءة ودقيقة.

الكلمات المفتاحية: صور النص الانكليزي، تمييز الانماط،، تميز الحروف، طريقة البلوك ، معالجات تعدد النواة.

Diyala Journal for Pure Science

Parallel Processing Based on Fragmentation of English Text Pictures to Speed Up Recognition of Characters on a Multi-Core Computer

Maisa'a Abid Ali and Abdul Monem S. Rahma

Introduction

Parallel computing is simultaneous operation to utilize of different computer resources to solve a computational task. The major precept of parallel computing is a split a task in such as a method that the task implements in minimum time with maximum efficiency [1]. A shared memory multi-central processing unit through connecting multiple-central processing units to one memory system. To perform parallel computing there can be several of kinds of parallel instrument, such as a cluster of computers combined with a joint together with a high-velocity network [1].

Pattern recognition is interested in the designing and evolution of systems that recognize patterns in datum. The objective of a pattern recognition program is to analysis a scene in the actual world and to reach at a detailing of the scenery which is helpful for the achievement of some task [2]. The actual world monitoring is combined out of searchlight and pattern recognition system classify or depict these monitoring. A characteristic extraction technique computes numeric or symbolic data from these monitoring [3 and 4]. The designing of a pattern recognition system basically includes the following three-part: datum impersonation, Classification and lastly, Prototyping. A lot of jobs have been done concerning document pictures noise revelation and elimination but most of them deal with unstructured noise or chaos on documents with easy surroundings [5 and 6].

Related Work in Pattern Recognition

The domain of pattern recognition is needed for document analysis. Therefore, more researchers have worked in this domain recently.

In 2010. Lukas Neumann, and Jiri Matas. proposed manner is new as it:

1. Departs from strict feed-forward pipeline and replaces it by a hypotheses verification framework simultaneously processing multiple text line hypotheses.

2. Uses synthesis fonts to train the algorithm eliminating the need for time-consuming acquisition and labeling of real-world training data.
3. Exploits maximally stable extremal regions ($MSER_s$) which provides robustness to geometric and illumination conditions.

The outcom of the first to report both text detection and recognition for using dataset of Chsr 74 k, the recognition rate is 72% is a chieved, and 18% higher than the state-of-the-art. The text localizationworks for number of alphabets and the method is easily adapted to recognition of other scripts [7].

In 2013, Alvaro et al. proposed defining a bidimensional extended of stochastic context-free regulation for sheet segment of build paper. Two collections of script classified feature is using to perform an initial classification of each zone of the sheet. The sheet is segments to obtain the most from hypothesis according to the grammar. This method shows significant improvement in several cases compared to using conditional random fields. Regulation supply a given segment that allows a semantic estimation, which also establishes this model [8].

In 2013, Cirera et al. proposed a hybrid speech type for recognizing handwritten historic paper with a build syntactical mode. Using a hidden Markov pattern-based recognition framing, word-based regulation with a closed thesaurus is enhanced by a character-sequence recognition method. This allows recognizing thesaurus words in planned partitions of the recognition while maintaining a closed vocabulary limitation for another partition. The stream situation of the project is a job in forwarding, and it can report improvement in terms of the symbol mistake rate [9].

In 2015, Maclean and Labhan. proposed recognizing handwritten mathematics, which is a defy rating issue, demand simultaneous consistency of all the character include the input and compound 2D relationships among character and subexpressions in the handwritten work. It can be current a system that catchall recognisable translation of the input and organizes them in a parse forest from which individual parse trees may be taken away and reported. The tree

extraction stride uses a novel probabilistic arbour scoring organization in which a Bayesian network is structured based on the structure of the input, and each joint variable task matching to a various parse tree depending on probability.

The outcome of accuracy demonstrates that the performing recognition system is more accurate than the previous versions (which used a non-probabilistic way) and another theoretical mathematics recognizers [10].

Multi-core Computers and Parallel Processing

Parallel processors are computer systems consisting of multiple processing units connected via some interconnection network to make more CPUs available to modify the programming method to obtain high efficiency. These are two main agents utilized to categorize such as systems: the processing units themselves, and the correlation network that equalize them together. The processing units can connect and react with every another utilize either shared memory or letter-crossing techniques.

The connection network for shared memory systems can be categorized as carriage-based on against transition-based networks. In letter-crossing systems, the connection network is partition into "static and dynamic" connection. Static connections have a constant topology that does not changes whilst programs are executing. Dynamic connections inspire links on the fly such as the program execution [11 and 12].

The major excuse for utilizing multiprocessors is to differentiate strong computers via easily communication multiple processors. A multiprocessor is predictable to arrive quicker speed than the quicker one-processor system. In addition, a multiprocessor consisting of an issue of one-processors is predictable to be more set back-touching than structure a high-performance one processor [13 and 14].

Pattern Recognition

Pattern recognition can be known as the categorization of insert datum into identifiable share through the extracted of important characteristic or attribute of the datum from the surroundings of irrelevant detail [15]. Some researchers knew it as a domain interested with instrument recognition of meaningful regularities in noisy or difficult environments. Pattern recognition is a generic term to specify a wide range of problems such as recognition, describing, classified, and grouping of patterns [15].

The design of a pattern recognition system substantially includes the followed three aspects: datum represented, Classified and in final, Prototype. The problem area grammar the choose of the sensor, pre-processing technique, represented scheme, and decision- making models [16]. The recognition of a word involves the pattern, pose, matching, identification, or object discrimination. Nevertheless, an efficient method of dealing with recognition is through an efficient representation model [17].

Proposed System

This proposed system offers a new algorithm for pattern recognition in parallel processing multi-core computers for English language script images and uses four central multi-core processing units **(**CPU1, CPU2, CPU3, and CPU4). The image analysis in pattern recognition of a word in the English language can be divided into a set of blocks in each row for the English text image analysis. The system implementation has four steps, and each step is executed in one processor. In other words, each row is executed in one processor fromCPU1, CPU2, CPU3, and CPU4. English system rely on number of main steps of execute system, first step: is load English script of image file for number of pages (from 6 to 150), second step: is partition image file to several image pages (from 6, to 150), third step: is divide the image into number of blocks for each pages (from 6 to 150) in second step, and final step: can be divided each pages in third step language scripts in image recognition operation are selected in a sequence row by row and use shared memory operations in each row, divided into a set of blocks. Each block

consists of a number of character recognition for one word in one block and the calculated the accuracy of each word recognition and total time in each processor. The sixth section tests of the results explain the operation of recognition in parallel processing. Figure 1 illustrates the general flowchart of a new algorithm for the system in each central processing unit (CPU1, CPU2, CPU3, and CPU4).

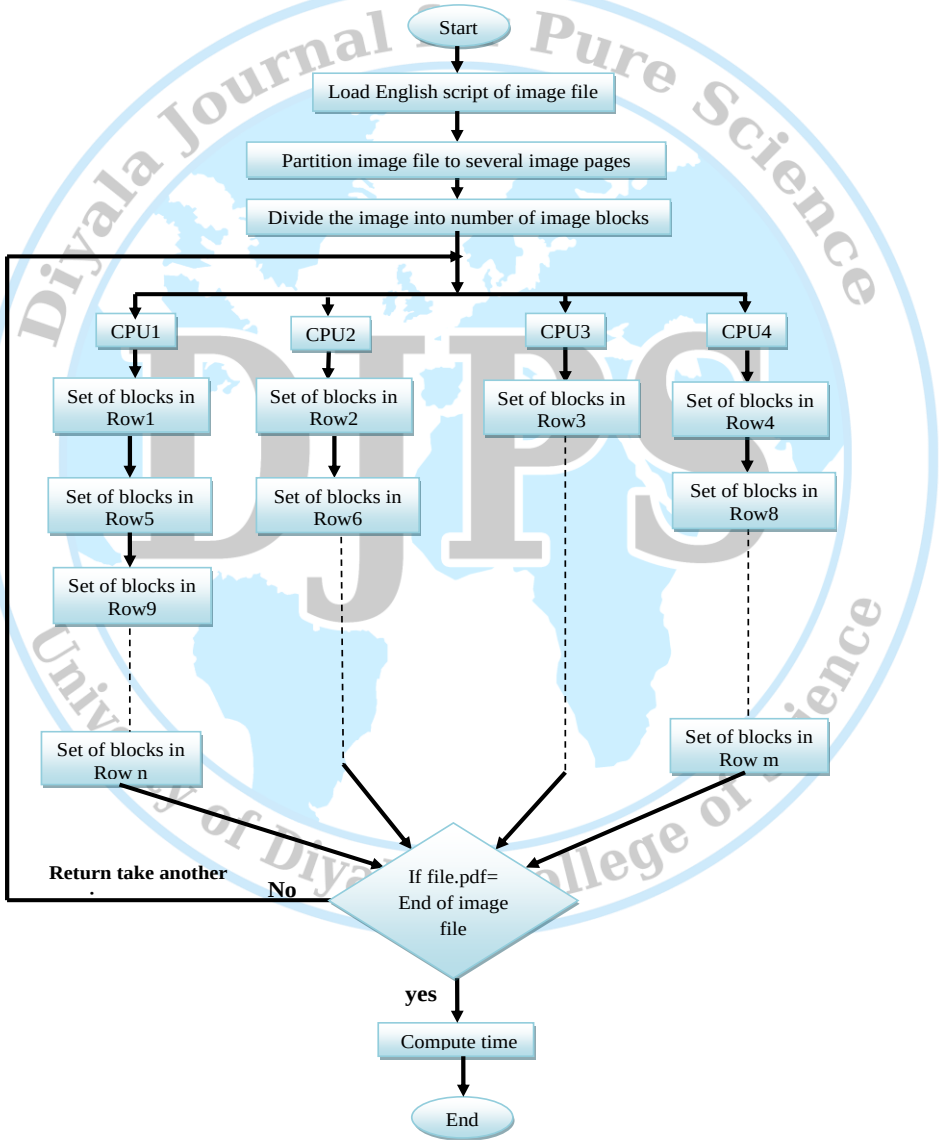

Figure 1: General flowchart of algorithm for the system.

Implementation of the Recognition Algorithm on One Processor

This algorithm recognizes words in English scripts in images and computes the total time for one processor using shared memory. The detail of the algorithm is explained below.

Algorithm for recognizing words in English scripts in images in a one-core processor

Process: Algorithm for recognizing words in English scripts in images in a one-core processor.
Input: English scripts in images. Output: Recognition words in file.pdf.
Initial: A: Load script image, B: Partition image file, C: Split script images in one processor p1 for n blocks in each row, D: Recogniseeach word in the block in each row, E: Compute the total time in one processor, F: Load each recognition word in file.pdf Step 1: Load script image into A. Step 2: Partition image file to image pages in B. Step 3: Split script images into the shared memory, divided into on blocks in each row into C. Step 4: Recognise words in blocks into D. Step 5: Condition if file.pdf not equal, end image file, return to Step 3. Step 6: Compute total time for one processor P1 into E. Step 7: Put the result of recognised words in file.pdf into F. Step 8: End.

Implementation of the Recognition Algorithm on Two Processors

This algorithm recognizes words in English scripts in images and computes the total time using two processors using shared memory in each processor. The detail of the algorithm is explained below.

Algorithm for recognizing words in English scripts in images in a multi-core (two) processor

Process: Algorithm for recognising words in English scripts in images in a multi-core (two)processor.
Input: English scripts in images. Output: Recognition words in file.pdf.

Initial A: Load script images, B: Partition image file, C: Split script images in processor P1 andP2 for n blocks in each row, D: Recogniseeach word in the block in each row, E: Compute the total time with two processors, F: Load each recognition word in file.pdf Step 1: Load script images into A. Step 2: Partition image file to image pages in B. Step 3: Split script images to shared memory in P1, and P2 is divided for n blocks in each row into C. Step 4: Recognise words in blocks in two processors, P1 and P2, into D. Step 5: Condition if file.pdf not equal, end image file, return to Step 3. Step 6: Computed total time in two processors, P1 and P2, into E. Step 7: Put the result of recognized words in file.pdf into F. Step 8: End.

Mplementation of the Recognition Algorithm on Three Processors

This algorithm recognizes words in English scripts in images and computes the total time using three processors using shared memory in each processor. The detail of the algorithm is explained below.

Algorithm of recognizing words in English scripts in images in a multi-core (three) processor

Process: Algorithm of recognizing words in English scripts in images in a multi-core (three) processor.
Input: English scripts in the image. Output: Recognition words in file.pdf.
Initial A: Load script images, B: Partition image file, C: Split script images into processorsP1, P2, andP3 for n blocks in each row, D: Recognise each word in the block in each row, E: Compute the total time in two processors, F: Load each recognised word in file.pdf Step 1: Load script images into A. Step 2: Partition image file to image pages in B. Step 3: Split script images into shared memory in P1 and P2, and P3 is divided for n blocks in each row into C. Step 4: Recognise words in blocks in three processors P1, P2, and P3 into D. Step 5: Condition if file.pdf not equal, end of image file, return to Step 3. Step 6: Computed total time in three processors, P1, P2, and P3 into E. Step 7: Put the result of recognized words in file.pdf into F. Step 8: End.

Implementation of the Recognition Algorithm on Four Processors

This algorithm recognizes words in English scripts in images and computes the total time using four processors using shared memory in each processor. The detail of the algorithm is explained below.

Algorithm of recognizing words in English scripts in images in a multi-core (four) processor

Process: Algorithm of recognizing words in English scripts in images in a multi-core (four) processor.
Input: English scripts in images. Output: Recognition words in file.pdf.
Initial A: Load script image, B: Partition image file, C: Split script image in processor p1, p2, p3, and p4 for n blocks in each row, D: Recognise each word in the block in each row, E: Compute the total time in two processors, F: Load each recognised word in file.pdf. Step 1: Load script image into A. Step 2: Partition image file to image pages in B. Step 3: Split script image to shared memory in P1, P2, and P3, and P4 is divided for n blocks in each row into C. Step 4: Recognise words in blocks in four processorsP1, P2, P3, and P4 into D. Step 5: Condition if file.pdf not equal, end image file, return to Step 3. Step 5: Computed total time in four processors P1, P2, P3, and P4 into E. Step 6: Put the result of recognized words in file.pdf into F. Step 7: End.

Test and Results

This section computes the test of the results for the total time executed in the multi-core processors (from 1 to 4 processors).

Figures 2 and 3 show that the paper is six pages of English script in an image, for example six page and also test number of pages from 20, 40 ,60, 100, and 150 pages, given good result in performance system. whereas, increased number of pages in pattern recognition increased total time of recognition for words in image in multi-core

Diyala Journal for Pure Science

Parallel Processing Based on Fragmentation of English Text Pictures to Speed Up Recognition of Characters on a Multi-Core Computer

Maisa'a Abid Ali and Abdul Monem S. Rahma

1.1 Introduction:

Generally frame based method has many shortcomings in applications of Arabic linguistic steganography, especially kashida and single double quotation, such as easy visual detection by human eye, easy attack by applying copy attack, transform attack also problems related to the text cover size of steganography and capability of hiding data in text document is very difficult. This dissertation offers problem solution by hiding in protocol in Arabic scripts, using the embedded and extracted algorithms.

The solution for this problem is by using linguistic steganography methods and search method to find position to hide secret message without being sensed by attacker.

Information hiding is used to design framework from different eight methods to hide inside text document, and process the problem through these methods to obtain security, transparency (visible or invisible), robustness, and irreversible, by using the secret key generation for DRLR, RSVD, and STRI.

1.2 Steganography

Steganography is of Greek original meaning "hidden writing" from the Greek words stegano meaning "covered or protected", and graphics meaning "writing or drawing" [1].

Steganography is a branch of data hiding and its major goal is to communicate or transit the data security in a complete undetectable method [2], [3].

1.3 Linguistic Steganography

Linguistic steganography is focused on applying changes to a cover text so as to embed secret message, in a way that the changes do not cause any unnatural or ungrammatical text. According to cover text, linguistic steganography can be classified into two categories: generation based such as in [4], or transformation based such as in [5]. Linguistic steganography, as depicted in Figure (1-1), is one of steganography main branches, besides technical steganography, and digital steganography [6], [7],which consists of two types:

1. Semagrams: This hides information by using of sign or symbols. There are two types of semagrams: *Visual semagrams*: the visual semagrams use ordinary physical objects, above suspicion viewing to carry a message for example doodles, components positioning on website, desk [6], and *Text Semagrams*: the text semagrams hides a message by modifying the carrier text appearance for example subtle alteration in font type or size, extra space addition, varied trappings in hand written text, letters [6].
2. Open codes: hide a secret message with a legitimate carrier message in such a way that is obvious to an unsuspecting. There are two types of Open code: *Jargon code*: this uses languages that are understood by a group of people but an meaningless to other people, and *Covered Ciphers*:

1.4.3 Public Key Steganography

Public key Steganography doesn't rely on the interchange of a secret key. It demands two keys, one of them is special (secret) and the other is general: the public key is stored in a public database, whereas the public key is used in the embed operation. The secret key is utilized to rebuild the secret message. One path to construct a public key Steganography system is to utilized a public key crypto system. The transmit and the receiver can interchange public keys of several public key cryptography algorithm before imprison. Public key Steganography utilizes the fact that the decoded function in a steganography system can be applied to any cover, whether or not it already contains a secret message [8].

The public key steganography depends on the actuality that encrypted data is randomly suitable to hide in plain sight.

The transmit encrypts the data with the recipient public key to obtain a random-looking message and embeds it in a channel which defines the recipient, that way changes some of the natural randomness with which every communication process is joined. It is presumed that both the cryptographic algorithms and the embed functions are general define [9],[10].

1.5 Literature Survey

In the following review, different manners are used in order of linguistic steganography text.

1- In 2007, A. Gutub, and M.Fattani, introduced a novel Arabic text steganography technique for Arabic script using letter points and kashida.

The technique hides secret information as bits in Arabic letters (cover) by using kashida and points of letters. The technique considers un-point Arabic letters followed by a kashida if the secret bit is (0), and point Arabic letters followed by kashida if secret bit is (1).

Their technique enhanced robustness and security but might have some limitations with capacity of the cover media if the number of secret bits of the secret information is large. This steganography technique is found to be suitable for other languages having similar script to that of Arabic for example Persian and Urdu [11].

2- In 2009, A. H. Fahd, et al, introduced improving security, and capacity for Arabic text steganography using kashida. The approach hides secret information as bits within Arabic letters (cover) by using kashida using three scenarios. The approach discusses maximum number of kashida letters that can be added to the Arabic cover word. Also the researchers evaluated the number of hidden bits that can be embedded in the carrier file and compared the results with diacritics, and kashida methods [12].

3- In 2009 Ammar Odeh and Khaled Elleithy, introduced an algorithm that has used some of general characteristics Arabic texts, that are explained as few vowel letters. Arabic Diacritics is a volitional uses ownership to conceal datum and minimizes the prospect of suspicions conceal. And it utilizes a rendering metrical content, the file volume before and after extra diacritics and possibility to conceal data together. And conceal data in various covering media which appear defy safety causes. The file volume and style alteration ought to grow the eventuality to detect and utilize tools of steganalysis and that ought to command for detect the conceal data [13].

4- In 2009, M. Hassan Shirali-Shahreza and Mohammad Shirali-Shahreza, introduced a novel manner to conceal data in Persian (Farsi) and Arabic languages in Unicode scale, to conceal script of two characters of 'Ya' (ی) and 'Kaf' (ک) used. The two characters of (ی) and (ک) have themselves shape however various codes which they utilized at the start or in the center of words. The major aim of this manner is perception translucence. It has a stellar perception translucence which causes the stego-text which the user sees alright like the main text [14].

5- In 2010, Adnan Abdul-Aziz Gutub, et al, introduced an improved Arabic text steganography technique for Arabic text using kashida. The approach hides secret information as bits within Arabic letters (cover) by using extension character (kashida). The technique considers one kashida if the secret bit is (0) and two kashidas if secret bit is (1) after any letter which can hold it. The finishing character is embedded just after the last bit of the secret information, then the kashida is embedded randomly to the rest text in order to enhance the security of the technique. Also their technique enhanced security, capacity and robustness for Arabic texts based on secure communication [15].

6- In 2010, A. Ali and F. Moayad, introduced Arabic text steganography technique for Arabic text using kashida with Huffman code. The approach hides secret information as bits within Arabic letters (cover) by using extension character (kashida), and compressed the stego file using Huffman code. The technique considers absence of kashida if the secret bit is (0) and one kashida if secret bit is (1) after any connected letters. Also their technique is applied to other Arabic texts than that based secure communication, with different document formats [16].

7- In 2010, Abhishek Parakh, and Subhash Kak, introduced a k-threshold computational secret sharing technique that distributes a secret S into shares of size |S|/k-1, where |S| denotes the secret size. This bound is close to the space optimal bound of |S|/ k if the secret is to be recovered from k shares. In other words, the technique can be looked upon as a new information dispersal scheme that provides near optimal space efficiency. The proposed scheme makes use of repeated polynomial interpolation and has potential applications in secure information dispersal on the Web and in sensor networks [17].

8- In 2012 Vasil Kolev, Katya Tsvetkova, and Milcho Tsvetkov, introduced virtual m×n picture A from inspected astronomical photo plateful via utilizing onward little into than in the main model. By utilizing degree of a model, k it delete the excessive data or fuss and used such as Wiener filter, when degree k < m or k< n.

Figure 2: Six pages in the image

Diyala Journal for Pure Science

Parallel Processing Based on Fragmentation of English Text Pictures to Speed Up Recognition of Characters on a Multi-Core Computer

Maisa'a Abid Ali and Abdul Monem S. Rahma

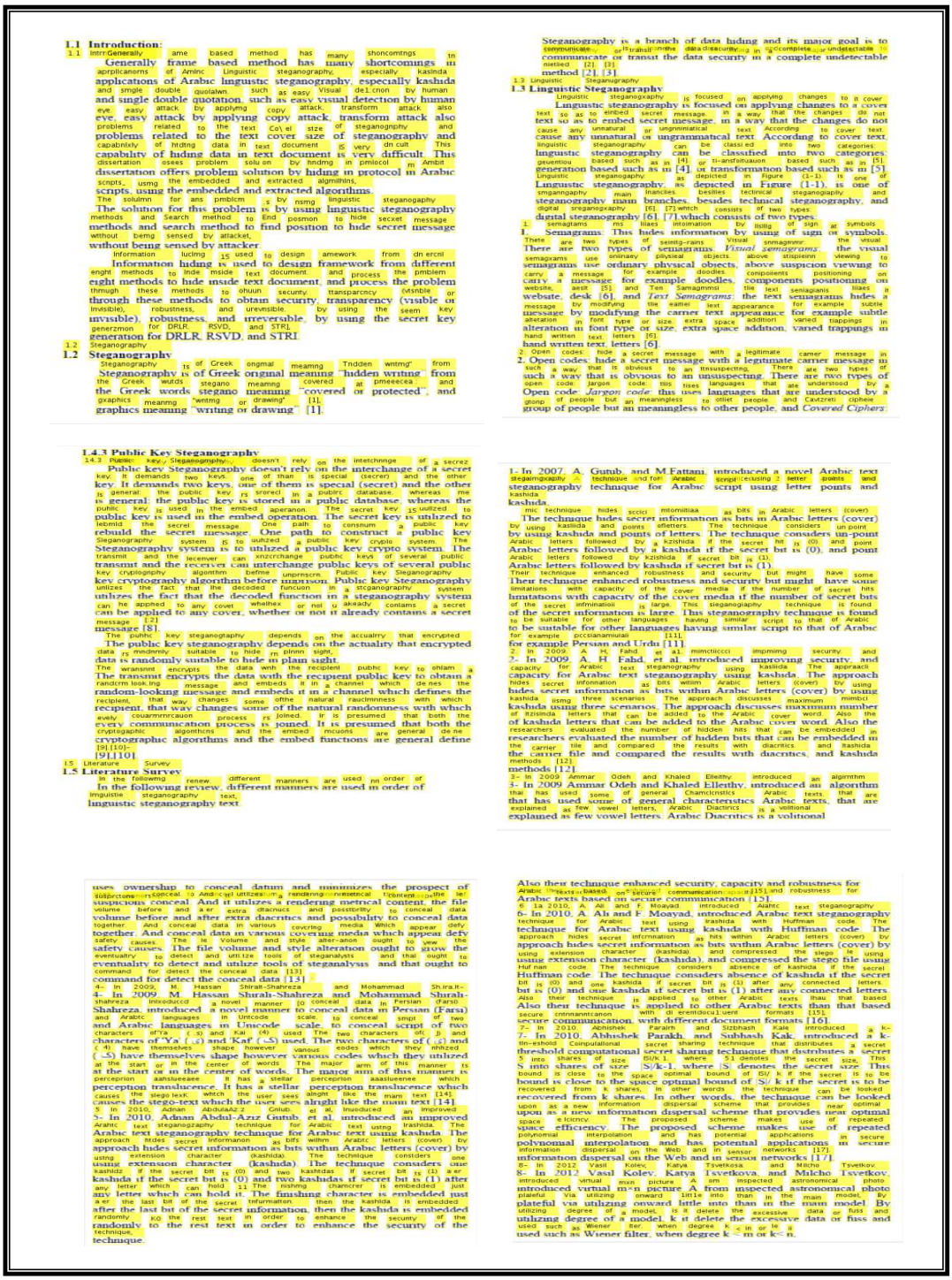

Figure 3: The six pages after recognition

Volume: 15, Issue:4, October 2019
Manuscript Code: 498B
P-ISSN: 2222-8373
E-ISSN: 2518-9255

Table 1 indicates the measure of the total time in the multi – core system. The total time varies for all four processors. Whereas the number of pages images in this table is indicate number of pages from 6 pages to 150 pages, it can test in proposed system. and can test for total time in millisecond for each processor in four fields in this table.

Table 1: Measure of total time executed in multi – core systems (in milliseconds)

Number of page images	Time for1CPU	Time for 2CPUs	Time for 3CPUs	Time for 4CPUs
6	38676ms	22338 ms	16892 ms	9669 ms
20	51.549 ms	30382 ms	20978 ms	12324 ms
40	62641 ms	34921 ms	24465 ms	17912 ms
60	82797 ms	43878 ms	28598 ms	20978 ms
100	95343 ms	58412 ms	33781 ms	24796 ms
150	110747 ms	70676 ms	40612 ms	29453 ms

Analysis Test and Result of Implementation

This section presents the analysis of the results of the recognition of words using the multi-core system. The computed total time in mill-second for each processor is shown.

Figure 4 illustrates the analysis of the multi-core systems in one CPUs for using number of image pages (1, 2, 3, 4, 5, and 6) is 6, 20, 40, 60, 100, and 150 perceptively pages for total time is calculated. Figure 5 presents the analysis of the multi-core system with two CPUs for using number of image pages (1, 2, 3, 4, 5, and 6) is 6, 20, 40, 60, 100, and 150 perceptively pages for total time is calculated. Figure 6 shows the analysis of the multi-core system using three CPUs for using number of image pages (1, 2, 3, 4, 5, and 6) is 6, 20, 40, 60, 100, and 150 perceptively pages for total time is calculated.

Figure 7 displays the analysis of the multi-core system using four CPUs for using number of image pages (1, 2, 3, 4, 5, and 6) is 6, 20, 40, 60, 100, and 150 perceptively pages for total time is calculated. The total time for the number of pages for file.docx or file.pdf is increased, this is relying on the file size whereas increased number of pages in file seen total time is increased.

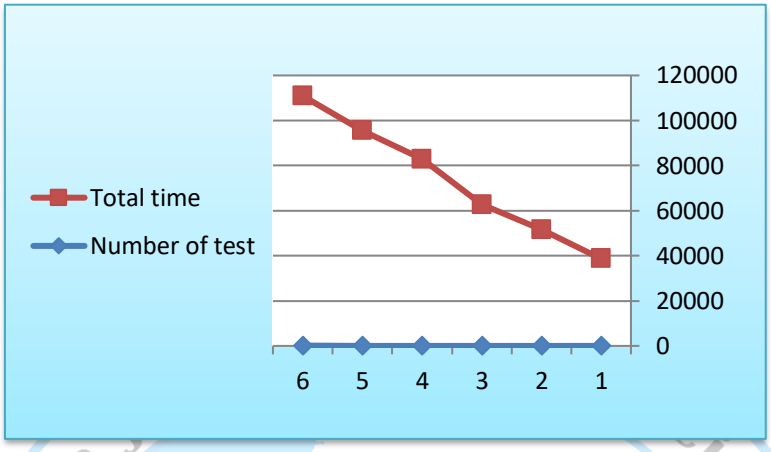

Figure 4: Total time for one CPU from 6-150 pages

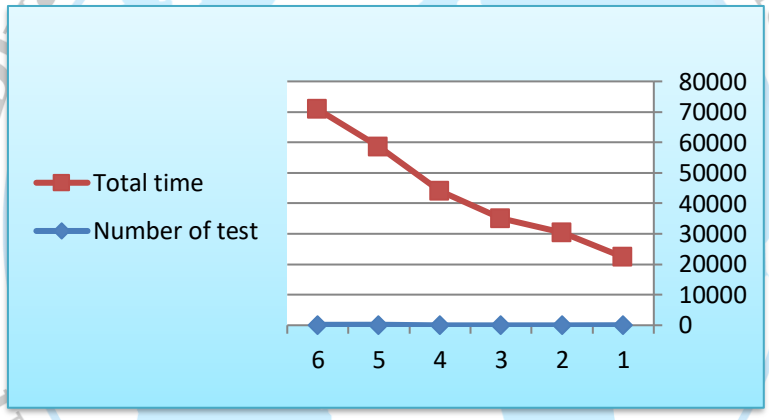

Figure 5: Total time for two CPUs from 6-150 pages.

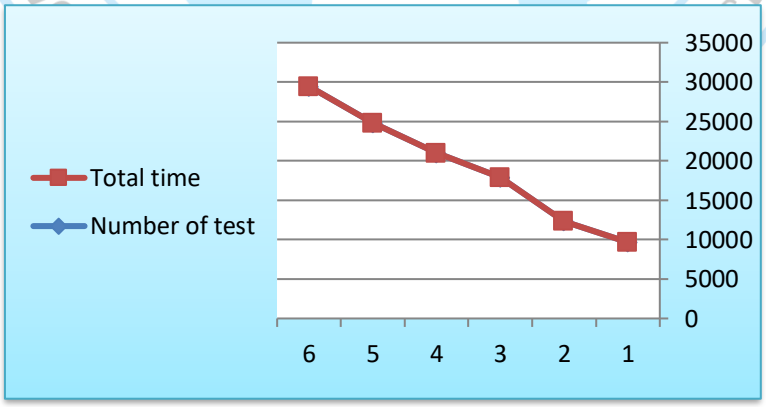

Figure 6: Total time for three CPUs from 6-150 pages

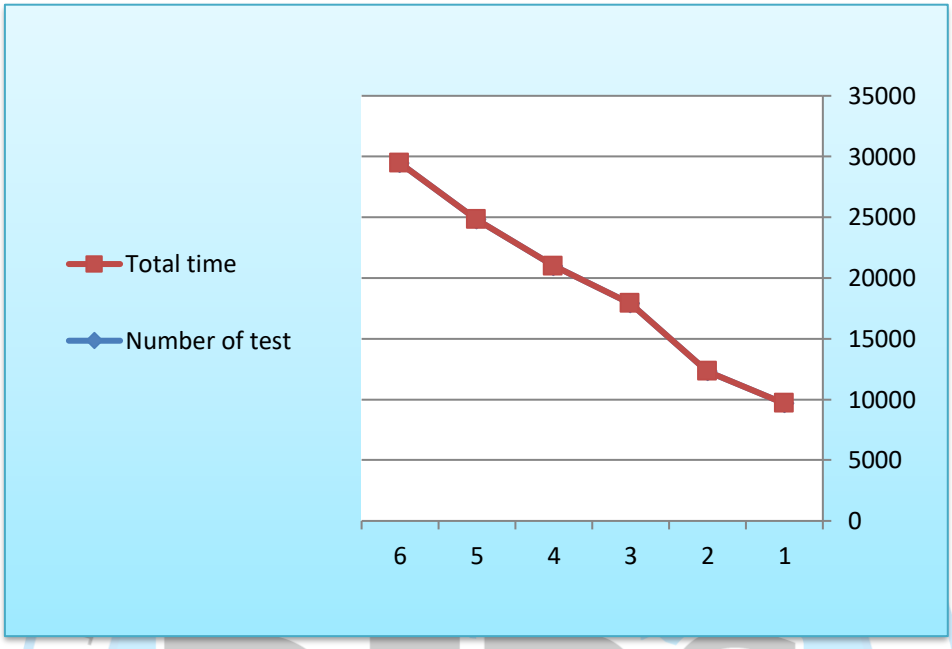

Figure 7: Total time for four CPUs from 6-150 pages

Conclusions

This paper focuses on the recognition of English words in images using multi-core computers. Our method is divided into multi-blocks of the script in images by row. This algorithm is good, efficient, robust, and fast for the recognition of words using four computers. This algorithm can be used for recognition of words across communication networks and the Internet. But previous work in this field not available on multi-processor system.

The outcome is obtained from the accuracy of the recognized words, which is fast and good. The text is saved in the text file 'file.txt' on a notepad, and the total time is calculated for each processor. The range of total time for one processor is from 38676 ms to 110747 ms. The range of total time for two processors is from 22338 ms to 70676 ms, the range of total time for three processors is from 9.669 ms to 40612 ms, and the range of the total time for four processors is from1617 ms to 29453 ms.

References

1. S. Saxena, S. Sharma, N. Sharma, Research Journal of Applied Sciences, Engineering and Technology 12(2), pp.223-238(2016).
2. R. Kasturi, L. O'gorman, V. Govindaraju, Sadhana, 27(1), pp. 3–22(2002).
3. S. T. Bow, Pattern Recognition and Image preprocessing, 2nd ed. (Marcel Dekker, new york, 2002).
4. T. S. Chernoy, V. M. Kliatskine, D. P. Nikolaev, A method of Periodic Pattern detection on Document Images, in: 29th European Conference on Modelling and Simulation (ECMS), 2015, Bulgaria, pp. 506-510.
5. K. Kpalma J. Ronsin, An Overview of Advances of Pattern Recognition Systems in Computer Vision, (hal. archives-ouvertes, 2007), https://hal.archives-ouvertes.fr/hal-00143842.
6. D. Sasirekha, E. Chandra, International Journal of Computer Applications 59(20), pp. 0975–8887 (2012).
7. H. El-Rewini, M. Abd-El-Barr, Advanced Computer Architecture and Parallel Processing, (John Wiley & Sons, Inc., Hoboken, NJ, 2005), p. 287.
8. P. Pacheco, An Introduction to Parallel Programming, (Morgan Kaufmann, Elsevier, 2011).
9. C. Breshears, The Art of Concurrency: A Thread Monkeys Guide to Writing Parallel Applications, (O'Reilly, Sebastopol, CA, 2009).
10. W. Pan, T. D. Bui, C.Y. Suen, Text Detection from Natural Scene Images Using Topographic Maps and Sparse Representations, In: IEEE ICIP. IEEE Computer Society, 2009.
11. C. M. Bishop, Pattern Recognition and Machine Learning, (Springer, 2006).
12. T.E. de Campos, B. R. Babu, M. Varma, VISAPP, (2009).
13. M. Adan, A. Adan, Solids Characterization Using Modeling Wave Structures, (Springer-Verlag Berlin Heidelberg, 2003).

14. N. Lukas, M. Jiri, A method for text localization and recognition in real-world images, In: 10th Asian Conference on Computer Vision, 8 (12) November (2010), Qeenstown, New Zeeland, pp. 1-14.

15. F. Alvaro, F. Cruz, J. A. Sanchez, O. R. Terrades, J. M. Benedi, Page Segmentation of Structured Documents Using 2D Stochastic Context-Free Grammars, In: Iberian Conference on Pattern Recognition and Image Analysis, 5-7 June (2013), Portugal, pp. 133-140

16. N. Cirera, A. Fornes, V. Frinken, J. Llados, Hybrid Grammar Language Model for Handwritten Historical Documents Recognition, In: Iberian Conference on Pattern Recognition and Image Analysis, 5-7 June (2013), Portugal, pp. 117-124.

17. S. Maclean, G. Labahn, Elsevier, Pattern Recognition, 48 (8), pp. 2433-2445 (2015).