Comparison of data recovery techniques on master file table between Aho-Corasick and logical data recovery based on efficiency

Hussein Ismael Sahib[1], Nurul Hidayah Ab Rahman[2], Ali Kazem Al-Qaysi[3], Mothana L. Attiah[4]
[1,2]Department of Information Security, Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, Malaysia
[3]Department of Telecommunications and information systems, Faculty of School of Computer Science and Electronic Engineering, University Essex, United Kingdom
[4]Department of Computer Engineering, Electrical Engineering Technical College, Middle Technical University, Baghdad, Iraq

Article Info

Article history:

Received Apr 7, 2020
Revised Jul 10, 2020
Accepted Aug 29, 2020

Keywords:

Accuracy
Aho-Corasick algorithm
Data recovery
Digital forensics
Logical algorithm
Master file table
New technology file system
Time

ABSTRACT

Data recovery is one of the tools used to obtain digital forensics from various storage media that rely entirely on the file system used to organize files in these media. In this paper, two of the latest techniques of file recovery from file system (new technology file system (NTFS)) logical data recovery, Aho-Corasick data recovery were studied, examined and a practical comparison was made between these two techniques according to the speed and accuracy factors using three global datasets. It was noted that all previous studies in this field completely ignored the time criterion despite the importance of this standard. On the other hand, algorithms developed with other algorithms were not compared. The proposed comparison of this paper aims to detect the weaknesses and strength of both algorithms to develop a new algorithm that is more accurate and faster than both algorithms. The paper concluded that the logical algorithm was superior to the Aho-Corasick algorithm according to the speed criterion, whereas the algorithms gave the same results according to the accuracy criterion. The paper leads to a set of suggestions for future research aimed at achieving a highly efficient and high-speed data recovery algorithm such as the file-carving algorithm.

Corresponding Author:

Hussein Ismael Sahib
Department of Information Security
Faculty of Computer Science and Information Technology
Universiti Tun Hussein Onn Malaysia (UTHM)
86400 Parit Raja, Johor, Malaysia
Email: eng.hussienissmail@gmail.com

1. INTRODUCTION

Data recovery algorithms play an effective role during digital investigative procedures so the data on computer hardware has become one of the most important criminal evidence sought by investigators, in the United States alone, the FBI reported over 300,000 complaints of online criminal activity in 2011 [1-5], costing nearly $500,000,000. Data recovery depends primarily on the file system used to store and format data in different volumes (new technology file system (NTFS), file allocation table (FAT), extension (EXT), ANDROID) [6-11]. All previous file systems do not delete data completely [12, 13]. In other word, file systems do not remove data from storage media because it is time-consuming [14]. In this research, the structure of the NTFS file system has been studied because of the system spreads around the world and the desirability of many users [15, 16]. The majority of the previous research is classified according to two categories: the first describes the structure of the NTFS file system, and the second type of research was as few as we mentioned.This work was found to provide a detail description of the structure of the NTFS file system, especially the master file table (MFT) structure [17], which is the heart of the NTFS file system and is the basis for file recovery for the target file system [18-20].

The previous studies were limited to study on the effectiveness of one algorithm only like Aho-Corasick data recovery algorithm [21, 22]. On the other hand, this algorithm the time criterion was neglected knowing that the speed of performance is one of the most important factors in the field of digital investigation [23]. While the logical data recovery algorithm [24], has adopted a single data set to study the effectiveness of the lgorithm and did not take into account overstatements such as the study of where a non-global data set consisting of nine files with varied types without overwrite and neglected the overwrite data. Moreover, this paper found a few studies that investigate data recovery algorithms and ignore the time factor (speed) in research available in this field. On the other hand, algorithms developed with other algorithms were not compared. In this paper, this study introduced a practical and effective comparison between data retrieval algorithms (data recovery using Aho-Corasick algorithm, logical data recovery algorithm) and three datasets (DFR-09 [25], T5 corpus without overwrite, T5 corpus with overwrite) based on the NTFS file system, to determine their efficiency and their ability to recover deleted data in the shortest possible time and with the greatest possible effectiveness from storage devices using visual studio C programming language. This comparison will clearly show the strengths and weaknesses of each of the algorithms presented, which is a key point in the development of these algorithms based on the time needed to restore data and the size of data restored.

2. RESEARCH METHOD

File recovery techniques and tools are many, varied and are constantly evolving. Each technique has weaknesses and strengths that distinguish them from the rest of the techniques. Therefore, there is a question that always asks which techniques should be used to restore data as quickly and efficiently as possible. This study showed the most important modern techniques and, in this section, in particular, a mechanism was developed to compare these techniques based on two main variables: speed and accuracy. Proposed method the general architecture of proposed method is illustrated in Figure 1.

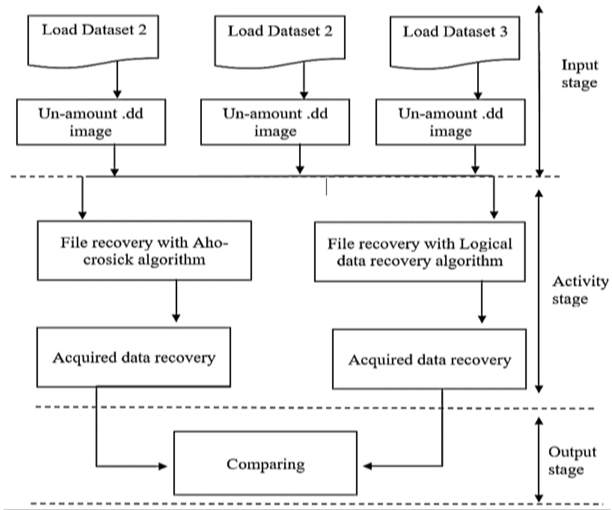

Figure 1. The general architecture of the proposed method

In Figure 1 we proposed our methodology to compare two algorithms according accuracy and speed that can be described in several steps as follows:

a. Input stage

During this phase, three different data sets will be loaded, and the contents and size of each data set will be recorded. This study, three data sets were selected that included different types of data in different sizes

(1st dataset it is DFR-09, 2nd dataset it is the T5 corpus without overwrite and 3nd dataset it is the T5 corpus with overwrite.

b. Activities stage

During this phase, a series of actions will be undertaken, including: the first, designing a program to restore files according to technique "file undeleted using the Aho-Corasick algorithm" using C programming language. And second, designing a program to restore files according to technique "logical data recovery algorithm" using C programming language. During this study, three data sets were selected that included different types of data in different sizes. Table 1 shows the three data sets and the sources of these data sets.

Table 1. Used datasets and its sources

Dataset index	Dataset name	Source
1st dataset	DFR-09	NIST (https://www.cfreds.nist.gov/dfr-test-images.html)
2nd dataset	The T5 corpus without overwrite	Govdocs (roussev.net/t5/t5.html) without overwrite
3rd dataset	The T5 corpus with overwrite	Govdocs (roussev.net/t5/t5.html) with over right

– DFR-09

DFR-09 is a dataset provided by NIST to test data recovery tools which include a large number of text documents, where according to the image layout document which describes the creation and final layout of each image, DFR-09 has 266 deleted files without overwrite there are 14 files without metadata and therefore cannot be found. The files to be not found are: xB0023-3.txt, xB0054-5.txt, xB0074-3.txt, xB0106-5.txt, xB0113-4.txt, xB0116-5.txt, xB0125-3.txt, xB0175-4.txt, xB0176-3.txt, xB0178-4.txt, xB0211-5.txt, xB0227-3.txt, xB0244-4.txt, xB0257-5.txt.

– T5 corpus without overwrite

T5 corpus is a dataset provided by Govdocs (Digital Corpora) which include 4457 multi-type files such as (hypertext markup language (HTML), portable document format (PDF), text, word document, power point document, excel document, image.jpg and image.gif) with total size 1.78 GB (1,911,662,784 Bytes) without overwrite, the deleted files description shown in Table 2.

Table 2. Deleted files description of T5 corpus without overwrite

Type of files	Number of files	Size
HTML	1093	68.4 MB (71,744,700 Bytes)
PDF	1073	603 MB (632,785,429 Bytes)
Text	711	234 MB (245,564,037 Bytes)
Word document	533	219 MB (230,552,622 Bytes)
Power Point document	368	351 MB (368,991975 Bytes)
Excel document	250	277 MB (290,944,862 Bytes)
Image.gif	67	13.9 MB (14,608,670 Bytes)
Image.jpg	362	53.8 MB (56,470,192 Bytes)

– T5 corpus with overwrite dataset

T5 corpus with overwrite this dataset includes as same as previous dataset 4457 multi-type files with total size 1.78 GB (1,911,662,784 Bytes) but here with overwrite 633 MB (663,995,699 Bytes) with 1271 index multi-type files such as (hypertext markup language (HTML), portable document format (PDF), text, word document, power point document, excel document, image.jpg and image.gif) the deleted files description shown in Table 3.

Table 3. Deleted files description of T5 corpus with overwrite

Type of files	Number of files	Size	Damaged files	
HTML	1093	68.4 MB (71,744,700 Bytes)	-	size
PDF	1073	588 MB (617,545,748 Byte)	-	-
Text	711	-	711	-
Word document	533	-	533	234 MB (245,564,037 Bytes)
Power Point document	368	351 MB (368,991,975 Bytes)	81	219 MB (230,552,622 Bytes)
Excel document	250	277 MB (290,944,862 Bytes)	-	77.6 MB (81,436,237 Bytes)
Image.gif	67	13.9 MB (14,608,670 Bytes)	66	-
Image.jpg	362	53.8 MB (56,470,192 Bytes)	307	13.9 MB (14,601,133 Bytes)

Third: application software designed on selected data sets. Forth: calculate the comparison variables (speed of recovery and rate of recovery) defined in the second section of this research for each dataset according to each algorithm. The practical procedures used during this research can be viewed through flowchart shown in Figure 1.

c. Output stage

During this phase, the two previous algorithms (file recovery with Aho-crosick algorithm and file recovery with Logical data recovery algorithm) of during previous datasets apply to each of the algorithms, each time the results record where the results sorte by type, then the recover files are checke to distinguish the operable files from the non-operable files. Will be compared according to the comparison variables to reach the fastest and most efficient algorithm.

3. RESULTS AND ANALYSIS

In this paper order to compare the performance of the algorithms, three global data sets were used specifically to study the effectiveness of data recovery tools, where overwrite cases have been taken into account, as described in the Table 4 summarizes the results of applying the two algorithms under study to the selected data sets. In order to compare the performance of the algorithms, three global data sets were used specifically to study the effectiveness of data recovery tools, where overwrite cases have been considered, as described in the following.

Table 4. Results summary and compare

Dataset	Aho-Corasick		Logical Recovery	
	Speed of recovery [B/sec]	Rate of recovery %	Speed of recovery [B/sec]	Rate of recovery %
DFR-9	281,73545	94	416,000	94
T5-corpus without overwrite	11,929,252.94	100	12,871,416.54	100
T5-corpus with overwrite	10,622,377	67.4	11,596,888.80	67.4

The first data set (DFR-9) with a deleted data size of 566,468 bytes, the Logical data recovery algorithm can retrieve 94% of the deleted data at a speed of 416,000 bytes in one second while the Aho-Corasick algorithm can restore the same percentage of deleted data (94%) but at a speed of up to 281,73545 bytes per second, which means that the Logical data recovery algorithm was twice the speed of the algorithm Aho-Corasick. On the other hand, in the application of the second data set (T5-corpus without overwrite) which has total size 1.78 GB (1,911,662,784 Bytes), the logical data recovery algorithm was able to recover 100% of the deleted data, i.e., restore all deleted data at a speed of 12,871,416.54 bytes per second while the Aho-Corasick algorithm was able to recover all deleted data but at 11,929,252.94 bytes in one second. Whereas, in the application of the third data set (T5-corpus with overwrite 633 MB (663,995,699 Bytes) with 1271 index (multi-type files) the Logical data recovery algorithm was able to recover 67.4% of the deleted data, at a speed of 11,596,888.80 bytes per second while the Aho-Corasick algorithm was able to recover 67.4% of the deleted data but at speed of 10,622,377 bytes in one second. As shown in Table 4 and Figures 2 and 3.

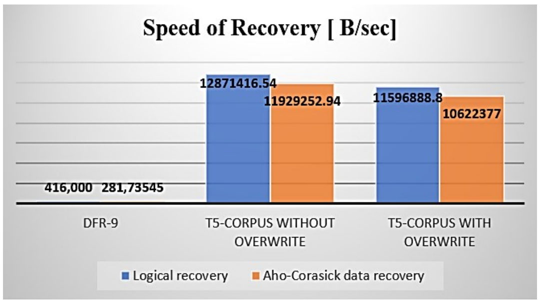

Figure 2. Chart speed of recovery [bytes/second]

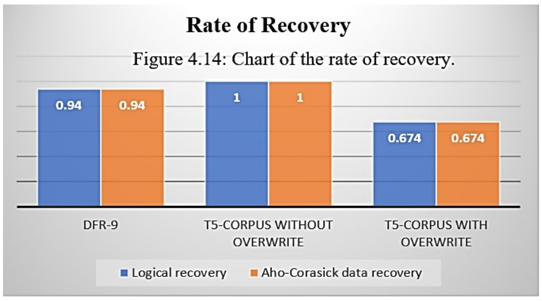

Figure 3. Chart of the rate of recovery

In the study of previous scientific research, all studies related to this field have omitted the standard of time required to recover files, while the study on the logical data recovery algorithm has neglected the overwrite data. Therefore, the focus of this research was on the time needed to recover data through the suggested speed criterion and to address the overwrite cases of deleted data. As can be seen in this research, the study of the effectiveness of two of the latest algorithms developed in the field of retrieving files deleted from the NTFS file system, while the previous studies were limited to study on the effectiveness of one algorithm only. On the other hand, the algorithms were studied according to the time and accuracy factors. The time criterion was neglected during the previous research, knowing that the speed of performance is one of the most important factors in the field of digital investigation. On the other hand, both of the two algorithms are unable to determine the undamaged file and restored only, where both of them recovered files destructive and non-readable. In addition to that both algorithms cannot determine if the file is completely destroyed or can retrieve information such as images or short audio clips or even text clips.

4. CONCLUSION

In this paper, three different data sets were used. As a result of the application of three different data sets, this work is looking into designing a highly efficient data recovery algorithm and higher speed than the algorithms being studied. The target algorithm begins by reading the MFT and specifies all metadata for each file has an index in the master file table. Then is the selection of the damaged files partially or completely. If restore all files was prompted, the logical algorithm will be followed. If we want to restore a specific type of data or a specific file, the Aho-Corasick algorithm will be used to search for and recovery the required files. On the other hand, the Aho-Corasick algorithm proved faster in searching and determining a specific type of file to retrieve. Therefore, the Aho-Corasick algorithm will be used to retrieve a specific type of data. In all cases, damaged files should not be restored which leads to waste in time. If the memory is formatted, the MFT will be empty and contain no information. The algorithms will fail to recover any file so that the algorithm should be able to retrieve data based on file structure such as file-carving techniques.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge UTHM Scheme, Universiti Universiti Tun Hussein Malaysia (UTHM), Department of information Security.

REFERENCES

[1] M. Patankar and D. Bhandari, “Forensic Tools used in Digital Crime Investigation,” *Indian J. Appl. Res.*, vol. 4, no. 5, pp. 278-283, 2014.
[2] S. Tomer, A. Apurva, P. Ranakoti, S. Yadav, and N. R. Roy, “Data recovery in Forensics,” *2017 Int. Conf. Comput. Commun. Technol. Smart Nation, IC3TSN 2017*, vol. 2017, pp. 188-192, 2018.
[3] M. Alhussein and D. Wijesekera, “A highly recoverable and efficient filesystem,” *Procedia Technol.*, vol. 16, pp. 491-498, 2014.
[4] Y. Yusoff, R. Ismail, and Z. Hassan, “Common Phases of Computer Forensics Investigation Models,” *Int. J. Comput. Sci. Inf. Technol.*, vol. 3, no. 3, pp. 17-31, 2011.
[5] P. A. S. Kapse, Priya S. Patil, “Survey on Different Phases of Digital Forensics Investigation Models,” *Int. J. Innov. Res. Comput. Commun. Eng.*, vol. 03, no. 03, pp. 1529-1534, 2015.
[6] F. Hafeez, “Role of File System in Operating System,” *International Journal of Computer Science and Innovation*, vol. 2016, pp. 117-127, 2016.
[7] M. Alazab, S. Venkatraman, and P. Watters, “Effective Digital Forensic Analysis of the NTFS Disk Image,” *Ubiquitous Comput. Commun. J.*, vol. 4, no. 3, pp. 551-558, 2009.
[8] S. Al-fedaghi and B. Al-babtain, “Modeling the Forensics Process,” *International Journal of Security and its Applications*, vol. 6, no. 4, pp. 97-108, 2012.
[9] M. Alazab and P. Watters, “Digital forensic techniques for static analysis of NTFS images,” *4th Int. Conf. Inf. Technol. ICIT*, 2009.
[10] K. L. Rusbarsky and K. City, “A Forensic Comparison of NTFS and FAT32 File Systems,” *Marshall Univ.*, p. 29, 2012.
[11] G. H. Fellows, “The joys of complexity and the deleted file,” *Digit. Investig.*, vol. 2, no. 2, pp. 89-93, 2005.
[12] J. Davis, J. MacLean, and D. Dampier, “Methods of Information Hiding and Detection in File Systems,” *2010 Fifth IEEE Int. Work. Syst. Approaches to Digit. Forensic Eng.*, vol. 5, no. June, pp. 66-69, 2010.
[13] C. Zoubek and K. Sack, “Selective deletion of non-relevant data,” *Digit. Investig.*, vol. 20, pp. S92–S98, 2017.
[14] S. Dillon, “Hide and Seek : Concealing and Recovering Hard Disk Data,” *James Madison University Infosec Techreport*, vol. 35, no. July, p. 17, 2006.
[15] G. S. Cho, “NTFS Directory Index Analysis for Computer Forensics,” *Proc. 2015 9th Int. Conf. Innov. Mob. Internet Serv. Ubiquitous Comput. IMIS 2015*, pp. 441-446, 2015
[16] K. Zhang, E. Cheng, and Q. Gao, “Analysis and implementation of NTFS file system based on computer forensics,” *2nd Int. Work. Educ. Technol. Comput. Sci. ETCS 2010*, vol. 1, pp. 325-328, 2010.
[17] NTFS, “NTFS Master File Table (MFT),” 2018. [Online]. Available: http://www.ntfs.com.
[18] R. R. Oommen and P. Sugathan, “Recovering Deleted Files from NTFS,” *EaseUS,* vol. 5, no. 5, pp. 2013–2016, 2016.
[19] C. K. Wee, “Analysis of hidden data in NTFS file system,” *Structure*, p. 9, 2005. [Online]. Available: http://www.forensicfocus.com/downloads/ntfs-hidden-data-analysis.pdf.
[20] S. H. Mahant and B. B. Meshram, “NTFS Deleted Files Recovery: Forensics View,” *IRACST-International J. Comput. Sci. Inf. Technol. Secur.*, vol. 2, no. 3, pp. 491–497, 2012.
[21] O. S. Sitompul, A. Handoko, and R. F. Rahmat, “File Reconstruction in Digital Forensic,” *TELKOMNIKA Telecommunication Computing Electronics and Control*, vol. 16, no, 2, pp. 776-794, 2018.

[22] O. S. Sitompul, A. Handoko, and R. F. Rahmat, “A file undelete with Aho-Corasick algorithm in file recovery,” *2016 Int. Conf. Informatics Comput. ICIC 2016*, no. lCIC, pp. 427–431, 2017.
[23] S. L. Garfinkel, "Digital Forensics," *American Scientist,* vol. 101, no. 5. 2013.
[24] P. Ravindra, R. Kalal, Soumya, and V. Mandal, “Logical data recovery technique for USB devices,” *Proc. - 2013 Int. Conf. Emerg. Trends Commun. Control. Signal Process. Comput. Appl. IEEE-C2SPCA 2013*, pp. 3-8, 2013.
[25] NIST, “DataSet-DFR-09,” [Online]. Available: https://www.cfreds.nist.gov/dfr-test-images.html.

BIOGRAPHIES OF AUTHORS

Hussein Ismael Sahib was born in Iraq, in 1984 He received the B.S Information Technology degree with honors from the Central Technical University, Baghdad, Iraq in 2006 and M.S Information Security degree from Universiti Tun Hussein Onn Malaysia (UTHM) in 2019. His research interests Digital Forensic & Information Security.

Nurul Hidayah Ab Rahman is currently an academic staff in the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia (UTHM). She received the PhD from University of South Australia, Australia.Her main areas of research interest are digital forensics analysis, information system security management, and security incident management.

Ali Kazem Al-Qaysi received the B.Eng. in Electrical Engineering from Faculty of Engineering, University of Tikrit, Iraq in 2004 and the M.Sc. Degree in Telecommunication Engineering from School of Computer Science and Electronic Engineering, University of Essex, United Kingdom in 2016.

Mothana L. Attiah was born in Iraq, in 1982. He received the B.S. degree with honors from the Middle Technical University, Iraq, in 2006 and M.S degree from Universiti Kebangsaan Malaysia (UKM) in 2016. He received the Ph.D. from the Faculty of Electronic & Computer Engineering, Universiti Teknikal Malaysia Melaka (UTeM). His research interests mainly focus on 5G mmWave Communications, spectrum sharing approach, & Network topology Planning and other topics related to security and data recovery.