Journal of Theoretical and Applied Information Technology
15th September 2018. Vol.96. No 17

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A REVIEW OF DIGITAL FORENSICS METHODS FOR JPEG FILE CARVING

RABEI RAAD ALI, KAMARUDDIN MALIK MOHAMAD, SAPIEE JAMEL, SHAMSUL KAMAL AHMAD KHALID

Faculty of Computer Science and Information Technology,
Universiti Tun Hussein Onn Malaysia, Johor, 86400, Malaysia
E-mail: rabei.aljawary@gmail.com, {malik, sapiee, shamsulk}@uthm.edu.my

ABSTRACT

Digital forensics is an important field of cybersecurity and digital crimes investigation. It entails applying file recovery methods to analyze data from storage media and extract hidden, deleted or overwritten files. The recovery process might have accompanied with cases of unallocated partitions of blocks or clusters and the absence of file system metadata. These cases entail advance recovery methods that have carving abilities. The file carving methods include different types of techniques to identify, validate and reassemble the file. This paper presents a comprehensive study of data recovery, file carving, and file reassembling. It focuses on identifying and recovering JPEG Images as it is a wildly covered in the literature. It classifies the carving techniques into three types: signature-, structure-, and content-based carvers. Subsequently, the paper reviews seven advanced carving methods in the literature. Finally, the paper presents a number of research gaps and conclude a number of possible improvements. Generally, both the gaps and possible improvements are associated with the fragmentation problem of data files.

Keywords: *Digital Forensics, Data Recovery, File Carving, File Reassembling, JPEG Image*

1. INTRODUCTION

Digital forensics offer an assistant platform for data discovery and analysis in crimes investigation to be used as evidence in the court [1], [2]. According to the 2014 United State Cybercrime Survey, 75% of the Internet users detect at least one security incident over a year. The Australian Computer Emergency Response Team (CERT) specify 56% of organizations out of 135 organizations expose to Cybercrime incident over a year [3].

Additionally, files might be accidentally lost or corrupted due to several reasons [4], [5]. Traditional file recovery software uses markers like headers and footers to identify and reconstruct the parts of a file. Current studies in digital forensic focus on improving carving methods accuracy in recovering files that having missing file systems or exposed to fragmentation. Mostly, these studies focus on carving JPEG file format [6]. The JPEG file format is widely used in computers, mobile phones, internet, multimedia applications, and digital cameras due to its sophisticated characteristics [7].

Subsequently, different JPEG file carving methods are proposed in the literature. Some examples are the work of Foremost [8], Scalpel [9], RevIt [10], PyFlag [11], Multimedia File Carver [12], myKarve [13], APF [14], and [15] and JPGcarve [16]. This paper reviews the literature on digital forensic and file recovery in general. The initial search is performed to build the review data which contains 98 papers, 13 theses, and three books. The sources of the data are a number of digital libraries and search engines such as Google Scholar, IEEE Explorer and etc. The final reviewed references that are closely related to the research topic are 61.

According to the review results, there are four recent review papers of files carving [2], [3], [17], [18]. This paper differs from the mentioned review papers in focusing only on JPEG file carving. It also differs in its classification and presentation to the carving methods and fragmentation cases. The paper farther reviews the Artificial Intelligence role in the carving process.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The following Section reintroduces digital forensics, file recovery, file carving and file carving categories. Section III focuses on JPEG file, JPEG file data format, JPEG file types and the thumbnail of JPEG file. Section IV presents the main carving techniques and their evaluation criteria. Section V explains the fragmentation types and cases of JPEG files. Section VI defines the reassembling and presents its related techniques. Section VII comprehensively review a number of carving methods and illustrates their main aspects and Section VIII presents the analysis of the carving methods and investigates the research gap. Finally, Section IX concludes the paper.

2. BACKGROUND OF FILE RECOVERY

2.1 Digital Forensics Investigation

Multimedia files play a basic role to support evidence analysis for making decisions about a crime via looking at files as a digital guide or evidence. Povar and Bhadran [19] define digital forensics as the practice of detecting, extracting, preserving, analyzing and presenting legally sound evidence of files guide derived from digital sources such as disk drives. Additionally, in the last few years, image sharing over the Internet (especially social media) has become more popular. Subsequently, the amount of shared illegal image has also increased, which cause the growth of cybercrimes [3]. Hence, increasing the importance of file recovery of digital forensic as supported by a cybercrimes report in risk survey.

There are many different pieces of data that can be preserved from image file [20]. In digital forensics, bit-copy images of disk drives are a common way for the process of investigation [21]. Digital forensics process starts with bit-copy images containing potential evidence with allocated and unallocated data. Allocated data can be defined as files containing active entries in the real file system metadata. It is easily accessible and can be recovered using traditional data recovery techniques. While unallocated data refers to unreferenced files by a file system information. Garfinkel [22] and Durmus et al. [23] define unallocated data as a data that its file system metadata is corrupted or missing. Figure 1 shows a general research map of file recovery.

Bodziak [24] state that criminals find new ways to hide their dirty work with increasing used digital device such as intentionally fragmenting a digital image. The fragmentation makes the files hard to be recovered. Additionally, over time the data present on hard drives continually changed when files are added, deleted or modified. Traces of this process can be seen in the unallocated data, which may contain non-fragmented files and fragmented files [25], [26]. Hence, the need for advanced forensic methods such as file carving that work completely independent from the underlying file system [27], [28].

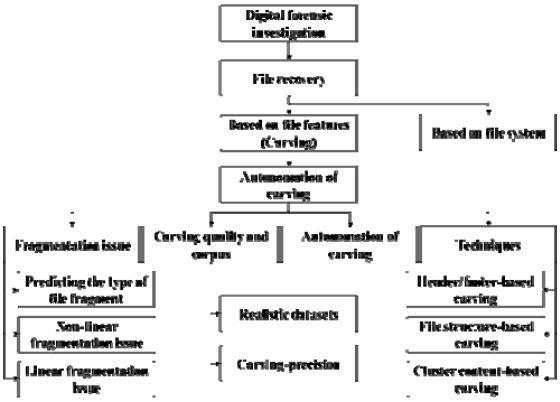

Figure 1: The research map of file recovery [17]

2.2 File Recovery Foundation

File recovery is the process of recuperating deleted or damaged files from the digital storage when their file system metadata is available [23], [25], [29], [30]. This is possible because of the hierarchical structure of most file systems such as FAT, EXT, NTFS, and HFS etc. where information about each file and its associated data might be still available [3]. [31]. Storages of computer devices are split into fix size storage units called sectors.

The file system groups these sectors into smallest allocation units called blocks/clusters that carries a data of a particular file [32]. The blocks/clusters of a file have markers that include a start of file "header" and end of file "footer" data known as a signature. The signature describes the file's system metadata of file type and contents [26]. Memon and Pal [14] define a cluster/block as the smallest area of storage that can be addressed uniquely. The deleted, damaged, or hidden file's information such as the information linking the clusters/blocks may still be available. Figures 2 shows different groups of data storage units.

Figure 2: Data storage units of computer storage [18]

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Thus, data recovery techniques can simply use file system metadata to recover damaged, failed or corrupted data from digital storage devices [28]. The file system is linked with the content section that holds information of clusters/blocks that are allocated to a particular file. Figure 3 shows an example of metadata entry where a deleted file is still existed and points to the blocks assigned to the current file. In case of a file system metadata is not available, then the recovery process needs more advanced techniques in order to recover deleted or hidden data [28], [33], [33]. In such situation, file carving process is important as an advanced method to recover files with a missing file system.

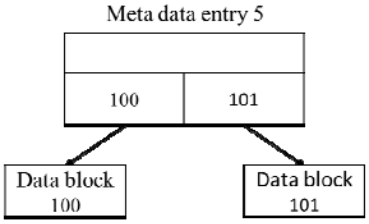

Figure 3: Metadata entry of a deleted file [36]

2.3 File Carving

File carving is the process of recovering deleted or damaged files from the digital storage when the file system metadata is unavailable [28], [33], [35]. Povar and Bhadran [19] define file carving as a process of retrieving files based on analysis of file contents including clusters/blocks or raw data. The carving method is utilized mostly for files that are available in an unallocated or a corrupted space where the file system is unavailable [28]. The traditional data recovery methods are fast and with high accuracy because they only work on the file system metadata. File carving is an important concept because it can handle all files even files with two or more parts.

Most file systems such as FAT, EXT, NTFS, and HFS etc. are affected by the problem of split files when files are expanded and modified then the file is stored on two or more locations [3]. The earliest carvers are simple header and footer carvers where these carvers simply search for file headers and file footers. If the header and footer are found, it extracts all the data in between them [11]. Later, file carving is capable of recovering a fragmented file in consecutive clusters locations [23], [33].

The Digital Forensics Research Workshop (DFRWS) frequently organizes a competition aiming to improve the carving methods [37]. The DFRWS is more focusing on developing fully or semi-automated carving techniques that are able to recover files or partial files with different fragmentation challenges. Figure 4 shows some examples of JPEG file with different fragmentation challenges.

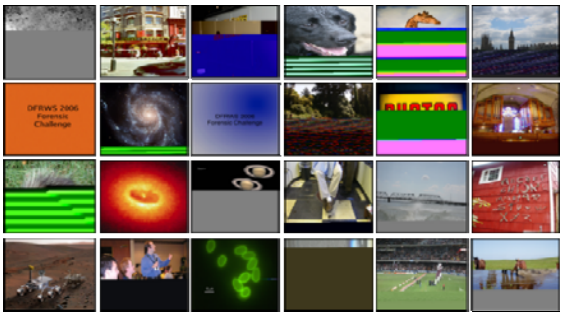

Figure 4: The DFRWS samples of a JPEG file

Fragmented file recovery is a difficult task as the fragmentation has multiple cases and many of which are still challenging to be completely solved [3]. The next section describes file carving categories and the extent to which they handle non-fragmented and fragmented files.

2.4 File Carving Categorizes

There are three categories of file carving techniques. The categorization is depending on the complexity of the recovery cases as described in [32]. The three categorize are listed in low to high complex ascending order as follows:

2.4.1 Signature-based Carving

Signature-based carving, it works by searching in dataset for the patterns that mark the beginning of a file (header) value (like FFD8 for JPEG files) and then looks for the first occurrence of the (footer) value (like FFD9 for JPEG files). All data clusters between the header and footer values are carved in an output file such as Scalpel carving technique [9]. Once the header value and end locations of a file are calculated, then all data clusters between these values are carved in an output file such as Foremost carving technique [8]. Header/maximum file size-based carving works similar to the header/footer carving, but it searches in the dataset for header value (like FFD8 for JPEG file) and then end of the file is educated guess or calculated based on its size. Both of the carving techniques assume that the header/footer data of a file are not missing (damaged or deleted) and non-fragmented.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2.4.2 Structure-based Carving

It is also sometimes referred as "Semantic carving" or "Deep carving" such as RevIt carving technique [10]. File structure-based carving first identifies a certain level of information in a file format. This information is matched in the raw data set to identify the file. This type uses the information of the internal file structure to carve a few fragmentation cases by reducing false positives of carving fragmented file.

2.4.3 Content-based Carving

The main idea behind cluster content-based is to read each individual cluster in the dataset and then analyze its contents to find out some relationships between the clusters that belong to a particular file. It calculates metadata information like character counts or statistical information over the bytes of the clusters. These clusters are later reassembled to recover the original file.

According to Garfinkel [38], two principal assumptions for content-based carving cluster/block which are (i) a cluster can only belong to a single file and (ii) a fragmentation can occur only at cluster boundaries. The difference in techniques lying under content-based carving is in the method used for analyzing the contents of individual clusters. Both of the above principles are required to parse each cluster and apply calculations to determine which file the cluster belongs to. These clusters then have to be reordered to recover original file. An example is the Adroit Photo Forensics (APF) work of [14] and [15]. The APF uses a metric for computing and measures the similarity of two clusters. Therefore, content-based carving techniques are found to be suitable for handling files with fragmentation conditions.

3. JPEG FILE

The Joint Photographic Experts Group (JPEG) is an international compression standard for continuous-tone still image for both grayscale and color types. This standard is designed to support ways to use high-quality graphics and pictures in digital devices [39]. Rao et al. [40] divide JPEG file formats to File Interchange Format (JFIF) and Exchangeable File Image Format (Exif). The JFIF is used for sharing in different applications via the Internet. The Exif is used for digital cameras and it embeds useful metadata relevant to a digital camera.

The JPEG standard has two basic compression methods [41]. The Discrete Cosine Transform (DCT)-based method is specified for lossy compression, and the predictive method is specified for lossless compression. The ways of displaying both lossy and lossless JPEG image on the screen is called modes of operation.

The JPEG image file has four compression modes: lossless mode, Sequential mode, Progressive mode, and Hierarchical mode. Figure 5 shows the relationship of major JPEG compression modes and encoding processes.

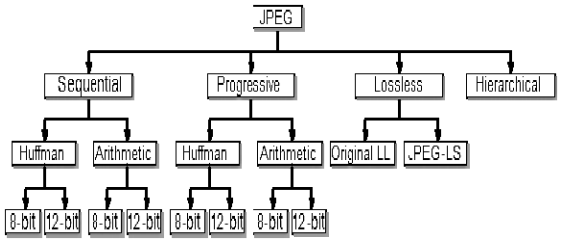

Figure 5: The modes of a JPEG compression [42]

The lossy technique (also is called baseline) is a DCT-based method that has been widely used in a large number of applications. The lossless mode uses a predictive method and does not have quantization process. The hierarchical mode can use DCT-based coding or predictive coding. The baseline JPEG system can have sequential, DCT-based or Huffman coding. The related limit of which is called the entropy rate [43]. The baseline system is shown in Figure 6.

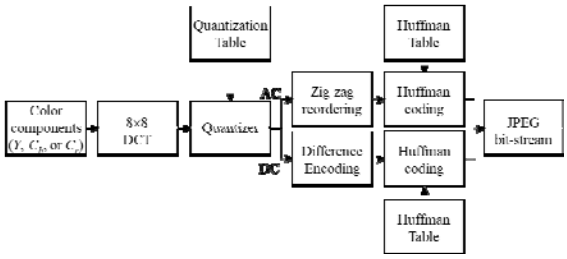

Figure 6: Baseline JPEG encoder [41]

The arithmetic coding and Huffman coding can be applied and their application depends on the imposed requirements. The arithmetic coding is recommended in the case of high-quality compression. The quantization and coding processes are controlled by parameter tables. The tables must be explicitly recorded in the code-stream, regardless of whether any attempt is made to optimize these parameters. Many compressors simply use the example tables described in the standard, with no attempt of customization.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Traditionally, a JPEG format consists of a single frame or a sequence of frames. Each frame is composed of one or more scans through the data, where a scan is a single pass through the data for one or more components of the image. At the beginning of each frame, the control procedures generate a frame header that contains parameter values needed for decoding the frame.

Similarly, at the beginning of each scan, the control procedures generate a scan header that contains parameter values needed for decoding the scan. Each frame and scan header start with a marker that can be located in the compressed data without decoding the entropy-coded segments. Marker segments defining quantization tables, entropy coding tables, and other parameters may precede the frame and scan headers. Figure 7 illustrates the structure of compressed image data stream.

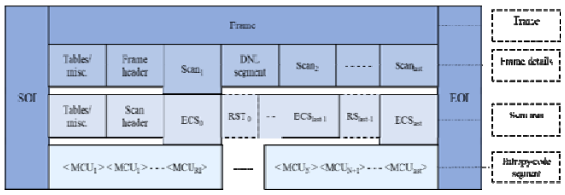

Figure 7: The JPEG data structure

The JPEG compressed data format contains three main parts, which begins with a start-of-image (SOI) marker, frame, and end-of-image (EOI) marker. There are two classes of segments of a JPEG format: marker segments and entropy coded segments. Marker segments contain header information, tables and other information required to interpret and decode the compressed image data, whereas Entropy coded segments contain the entropy coded [42]. Marker segments are always beginning with a marker, unique two-byte values markers that identify the function of the segment and distinguish various structural parts of the compressed data formats, a full list of these markers.

The structure of frame begins with the frame header and contains one or more scans. A frame header may precede by one or more table-specification or simultaneous marker segments. If a Define Number of Lines (DLN) segment is presented, it will immediately follow the first scan. For the DCT-based process, each scan contains from one to four image components. If two to four components are held within the scan, they will be interleaved within the scan. The scanning structure begins with a scan header and comprises one or more entropy coded data segments. Each scan header may precede by one or more table specification or miscellaneous marker segments. If Restart marker (RSTm) is not enabled, there will be only one entropy coded segment. If RSTm is enabled, the number of entropy-coded segments is defined by the size of the image and the defined restart interval. In this case, the RSTm follows each entropy coded segment except the last one.

The new JPEG formats allow multiple SOI/EOI, to support small images intertwined with the original image). Metadata often includes the APP segment marker "0xE1" in the header section to signify that original image includes another thumbnail in the standard JFIF/Exif [44]. The thumbnails are small images that are embedded in other images. Single or more thumbnails might fall into a JPEG image [45]. Birmingham et al. [37] propose thumbnails as a method to recover validated JPEG image in which each JPEG marker includes the size of the header that is identified by markers. They allow the algorithm to jump from header thumbnails to JPEG standard header until they find the valid header.

4. CARVING OF JPEG FILES

File carving of a JPEC file and other file formats, in general, is the process of reconstructing files without the aid of the metadata of file system data [46]. File system data are exposed to deletion for many reasons. The carving includes heuristics and probability handling algorithms that in order to successfully reassemble files [14], [47], [48]. File carving, in general, has three main procedures which are the identification of a file information, verification for the collected information of the file, and reassembly of the file [26], [47]. However, some fragments of the file might exist in unallocated memory, and hence advance carving methods are required to recover such files. Eventually, for efficient carving process rely on the identification of available information the heuristic of the reassembling algorithm. The following section, illustrate in details file carving of JPEG images.

4.1 The Carving Techniques

File carving is a technique that allows using information of internal file contents and file structure for recovering files from a corrupted hard disk or raw dataset and without file system metadata [49]. An important advantage of file carving is its ability to handle fragmented files [38]. The carving techniques use the operating system

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

strategy of saving files in consecutive clusters when it creates a new file or adds data to an existing file. In the case of there is no enough consecutive clusters, then the operating system splits and distributes the file to two or more parts.

The file carving techniques could recover a JPEG file fragmented the recovered JPEG file might be incomplete. This is because of the file carving techniques analysis a cluster/block or a set of clusters/blocks of a specific file format and its contents [33]. Some of the available data file recovery tools that implement carving techniques Foremost [8], Scalpel [9], RevIt [10], PyFlag [11], Multimedia File Carver [12], myKarve [13], APF [14] and [15] and JPGcarve [16]. There are some examples of implementing carving techniques in the literature.

Garfinkel [25] propose an object validation carver, namely Bifragment Gap Carving (BGC) to recover fragmented JPEG files. The carver validates whether clusters of a JPEG file belong to original JPEG files. Pal et al. [15] took validation further to propose Sequential Hypothesis Testing (SHT) using earlier work in Parallel Unique Path to be recovered as long as a sufficient validation function exists for the JPEG image. Cohen [11] describes advanced file carving technique by creating a JPEG image validator based on the open source libjpeg and a distance function to find sudden image changes, indicative of an invalid reconstruction.

4.2 The Evaluation of Carving Techniques

Kloet [50] describes two attributes for evaluating file carving techniques which are higher carving recall and higher carving precision. The higher carving recall is about detecting as much useful information as possible and do not simply discards any interesting results. Subsequently, the higher carving precision is about carving known false positives. Therefore, all fragmented file carving techniques should be able to recover all fragments to form complete and correct images. Eventually, high accuracy and performance remain the main issues of existing file carving applications.

In order to boost the improvement of file carving for the recovery of fragmented JPEGs, the Digital Forensics Research Workshop (DFRWS) initiated file carving challenges in both DFRWS (2006) [50] and DFRWS (2007) [51]. Researchers to validate their work intensively use these two datasets. The following sections succinctly describe the specific problems caused by fragmented files whereas Section V describes how different carving techniques (try to) overcome these problems.

5. THE FRAGMENTATION OF JPEG FILES

In order determine the quality of JPEG file carving results, the question that we set out is to answer why different techniques produce given different results for the same dataset. To answer this question, this section describes two types of JPEG files structures which are found in most of the datasets that are used to recover deleted files.

5.1 The Fragmentation Types of JPEG File

Usually, some of the JPEG files are continuously located on the disk drive and all clusters are adjacent to each other, due to shorter disk seek time. Figure 8 is a simple example of a strip in a disk with seven clusters. There are two types of clusters in this strip. The first type is a JPEG file cluster (J) and the second type is a fragmented cluster (F). the data might be stored in consecutive and contiguous clusters, of which J have 5 clutters' ((1,5), (2,5), (3,5), (4,5) and (5,5)), while (F) have 2 clusters ((1,2) and (2,2)). This pattern represents a consecutive and contiguous fragmented file.

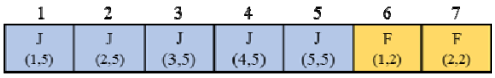

Figure 8: Consecutive and contiguous file clusters

File fragmentation is an obvious problem that affects many computers that are using a variety of file systems. Figure 9 is another simple example of a disk with seven clusters that illustrates how fragments occur. Nadeem Ashraf [32] defined the first generation of carvers used in file signatures or magic numbers to be matched with the file metadata for recover files in consecutive and contiguous files. Garfinkel [25] state that the fragmented files cannot find in most datasets but most of the files that are of interest to forensic investigations can be found in fragmented cases. Fragmentation occurs in an environment that is controlled by an operating system due to the circumstances [25]:

- Unavailable contiguous regions of free sectors.
- New data appended to existing files.
- Some file systems.

Figure 9 is a simple example of a strip in a disk with seven clusters. There are two types of clusters in this strip. The first type is a JPEG file cluster (J) and the second type is a fragmented cluster (F). The first clusters of the JPEG file J are (1,5), (2,5) and (3,5) and the second clusters of the JPEG file J are

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

(4,5) and (5,5). Subsequently, the fragmented clusters F are (1,2) and (2,2). This pattern represents a consecutive and non-contiguous fragmented file.

1	2	3	4	5	6	7
J (1,5)	J (2,5)	J (3,5)	F (1,2)	F (2,2)	J (1,5)	J (5,5)

Figure 9: Consecutive and non-contiguous file clusters

The next subsections succinctly describe the specific cases caused by fragmentation, identification of file clusters type and last subsection describes different reassembly techniques attempt to solve these fragments.

5.2 The Fragmentation Types of JPEG File

One important problem to deal with in file carving technique is the file fragmentation case. File fragmentation is an inescapable problem, which affects original drives using different file systems. A fragmented file can be divided into two categories, which are linear fragmentation and non-linear fragmentation [50], [53].

- Linear fragmentation occurs when a file split into two or multiple fragments, but the parts are present in the raw data in their original as shown in Figure 10.
- Non-linear fragmentation occurs when a file has been split into two or multiple fragments, but some of the parts are present in the raw data in a different order not as in the original order as shown in Figure 11.

Many cases may occur in linear fragmentation and non-linear fragmentation. The following and Figure 10 and Figure 11 shows six cases of linear and non-linear fragmentations:

Case1: JPEG file intertwined with the non-JPEG file [13].
Case2: JPEG file intertwined with JPEG file [35].
Case3: JPEG file fragmented with a gap between fragments [25].
Case4: JPEG file fragmented with contain missing fragment [54].
Case5: JPEG file fragmented into Multi-fragments [15].
Case6: JPEG file fragmented with a missing header [49].

	1	2	3	4	5	6	7
Case 1:	J (1,5)	J (2,5)	J (3,5)	F (1,2)	F (2,2)	J (1,5)	J (5,5)
Case 2:	J (1,5)	J (2,5)	J (3,5)	J (1,2)	J (2,2)	J (4,5)	J (5,5)
Case 3:	J (1,5)	J (2,5)	J (3,5)	G (1,2)	G (2,2)	J (4,5)	J (5,5)
Case 4:	J (1,5)	J (2,5)	J (3,5)	F (1,2)	F (2,2)	J **(4.5)**	J **(5.5)**
Case 5:	J (1,5)	J (2,5)	J (3,5)	(1,2)	J (4,5)	(2,2)	J (5,5)
Case 6:	J **(1.5)**	J (2,5)	J (3,5)	- (1,2)	- (2,2)	J (4,5)	J (5,5)

Figure 10: Examples of fragmented JPEG clusters with linear fragmentation

	1	2	3	4	5	6	7
Case 1:	J (1,5)	J (3,5)	J (2,5)	F (1,2)	F (2,2)	J (4,5)	J (5,5)
Case 2:	J (1,5)	J (3,5)	J (2,5)	J (1,2)	J (2,2)	J (4,5)	J (5,5)
Case 3:	J (1,5)	J (3,5)	J (2,5)	G (1,2)	G (2,2)	J (4,5)	J (5,5)
Case 4:	J (1,5)	J (3,5)	J (2,5)	F (1,2)	F (2,2)	J **(4.5)**	J **(5.5)**
Case 5:	J (1,5)	J (3,5)	J (2,5)	(1,2)	J (4,5)	(2,2)	J (5,5)
Case 6:	J **(1.5)**	J (3.5)	J (2.5)	- (1,2)	- (2,2)	J (4,5)	J (5,5)

Figure 11: Examples of fragmented JPEG clusters with nonlinear fragmentation

6. THE REASSEMBLING OF JPEG FILE

Recent advances in reassembling methods retrieve file fragments in fully- or semi-automatic recovery. Reassembling of the image file is used in several applied disciplines such as forensics, biology, archaeology and art restoration [14], [47], [55]. Although, many carving methods are used to reconstruction fragmented files [11], [25], [65]. Still, a few carving methods concentrate on reassembling fragmented files [33], [55], [56]. The reassembling of JPEG image entails identification to the various components of the reassembling image.

The identification helps to collect as much useful information as possible [57]. The reassembling methods rearrange the identified image components until it reaches its correct order. The following section illustrates the identification and reassembly methods of JPEG image.

6.1 The Identification of JPEG File

The identification in the reassembly involves recognizing clusters' details that belong to a specific file that is intended to be recovered. The

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

identification further improves the data for the reassembly step. There are three categories of files' type identification approaches. These categories are namely the extension-based identification, magic bytes-based identification, and content-based identification. Each identification approach has a number of advantages and disadvantages. Thus, none of the identification types are totally accurate and provide comprehensive solutions [55].

According to previous studies, many identification approaches solve the identification problems of binary and multi-classes such as decision trees, neural network, k-nearest neighbor, naive Bayes, software agent, and support vector machines [58], [59], [60]. However, there are a few attempts to classify high entropy file fragments in the literature [55], [61], McDaniel and Heydari [61] proposed Byte Frequency Analysis (BFA), File Header/Trailer (FHT) and Byte Frequency Cross-correlation (BFC) methods to analyze files' contents. Karresand and Shahmehri [63] have improved the Oscar method through measuring the Rate of Change (RoC) of the byte contents. However, statistical or non-statistical methods are required to analyze the extracted features [58].

Therefore, in order to improve the accuracy of the files identification and ultimately the reassembly process, effective and efficient identification methods need to be investigated. Qiu et al. [55] define reassembling as a process of detecting a fragmentation point of a fragmented file and then starting point of the following fragment. This process is repeated until a file is reconstructed or detained. Qiu proposes a new multimedia file carving method to enhance the recovery accuracy of fragmented files. The carving method includes extracting BFD and RoC features. They use Support Vector Machine (SVM) for the data files identification process. It is claimed that the identification step is able to improve the recovery accuracy of fragmented files and the JPEG files show higher recovery rates.

Veenman [64] propose a statistical learning approach for identification of fragmented clusters of different file types. The identification approach collects a number of evidence from the neighboring clusters for the purpose of setting characteristic features. Then it applies the statistical learning identifier on the extracted features to recognize files' patterns of relevant file types. The method targets files that are exposed to fragmented cluster in which both the header and footer of a cluster are not available. The method successively identifies file types of fragmented clusters.

In another study, Amirani et al. [61] propose a method based on a context-based file type identification. The method includes a Principle Component Analysis (PCA) technique for feature extraction and an unsupervised Multi-Layer Perceptron (MLP) neural network for identification. In this method, the extracted features are used to obtain a fileprint of each file of a type. The file prints are used to identify unknown files. The application of the method indicates high accuracy results and fast performance.

Zhang al et. [59] compare the performance of SVM and Extreme Learning Machine (ELM) with kernel functions when solving images recovery. The features are extracted from the ImageNet dataset. The results indicate that the ELM outperforms the SVM about 4% of the average accuracy. However, both the SVM and ELM are used for objects recognition purposes and not for recovering images of fragmented clusters.

6.2 The Reassembling Techniques of JPEG File

Clusters that have been identified according to their type are reassembled in their correct order. The outcome of this step is the original file or, in case of missing parts, a partially assembled file. There are three types of reassembling that have been discussed in [18]. They are file-signature, mapping function and graph types. The file-signature type is based on image contents such as header and footer markers or metadata that is included in files [29]. Garfinkel [25] find out that approximating 15-20% of larger files are fragmented and about 3% of files on disk can be recovered by assuming that they only consider of Bifragmentation (file fragmented into two fragments).

Therefore, Garfinkel [25] develop the Bifragment Gap Carving (BGC) technique which examines data between header and footer of fragmented files of DFRWS (2006) dataset for its validity in order to complete file. Cohen [11] formalizes file carving approach that examines the validity of reconstructed files using discriminators. The discriminators are characterized by their capability to verify the integrity of the files. Subsequently, Cohen proposes a mapping function that maps the offset within files by assuming that clusters of interest only belong to one single file. This approach cannot work well for JPEG image files, but it only can work for PDF and ZIP file types. These types utilize unique identifier objects that are implicitly managed by the files' structure.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Aronson and van den Bos [70] attempt to improve the accuracy of Cohen [11] approach.

Pal and his research group [14], [15], [33], [65] work on the graph type. The graph type uses a file carving strategy that realizes on identifying file types (e.g., marker sequences of a JPEG image) and semantic information (e.g., the color values of a fragmented JPEG file) to reassemble their original file [15]. Pal and Memon [33] interduce the Parallel Unique Path (PUP) algorithm for the reassembly of a graph file carver. The PUP calculates a weighting of file fragments as a metadata for reassembling. Figure. 12 presents the mean steps that are needed to operate on the scan area of JPEG files.

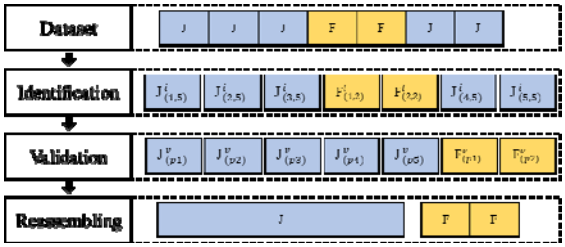

Figure 12: File carver architecture of [14]

Metz and Mora [10] introduce the Deep Carving algorithm. The algorithm is applied in the RevIt file recovery tool to perform the reassembling of file carving. The RevIt tool is tested in the DFRWS-2006 dataset. Garfinkel [25] introduces the Bifragment Gap Carving (BGC) technique. The BGC reassembles files that gaps of two fragments. Garfinkel further interduces the fast object validation technique to check the appropriateness of a file information before proceeding with the reassembling of the file carving. The validation process in different cases represents the conditions fulfillment of the reassembling or carving enrolment [16]. The next section describes a related work of the file carving techniques by gives an idea of their effectiveness, efficiency and the scope to which they handle fragmented JPEG files.

7. FILE CARVING METHODS

This section describes the related work that focused on presenting the current state-of-the-art file carving methods. Richard and Roussev [9] work establishes the foundation of file carving. They introduce Scalpel as an improvement of Foremost 0.69 [8]. The goal of Scalpel is to enhance the recovery performance and decrease memory usage. Both Scalpel and Foremost do not address the problem of fragmented files and carver only contiguous or non-fragmented files. Richard al et. [29] propose what they call “in-place file carving”, which is a method to reduce the requirements of hard disk space during file carving. Few other file carving methods are available in the literature. The recent methods (from 2006 to 2016) are analyzed and discussed in the following section.

7.1 RevIt

Metz and Mora [10] presented an open-source implementation of the smart carving and deep carving techniques as part of their contribution to the Digital Forensics Research Workshop (DFRWS) challenge, namely RevIt. These techniques allow the carver to look deeper into embedded files, probability factors, file types and features such as entropy, keywords, and fingerprints before applying the main carving technique. It aims to reduce the number of false positives which in turn will reduce time wasted and also storage needed. It carves files based on header, footer, content, section, and cluster and a verification function to cluster size. RevIt supports recovering from JPEG image file format. It uses DFRWS-2006 datasets for testing as the two datasets cover the testing the contributions of carving files. However, RevIt does not address the issue of reconstruction of a partial image and intertwined fragmented files.

7.2 Bifragment Gap Carving (BGC)

Garfinkel [25] propose the Bifragment Gap Carving (BGC) file recovery method. The BGC has a fast object validation technique for recovery of fragmented JPEG files. This BGC method recovers a file that has header/footer, can be decoded and validated and have two fragments (Bifragmented). The BGC identifies header/footer and exhaustively search all possible combinations of clusters between them until the decoding and validation conditions are successful.

Decoding condition means transforming the JPEG file information along with the cluster’s data into their corresponding format before processing by the carver. Validation condition means that the structure of a file meets the standard file structure of a file type. A validation method checks the structure of each cluster and ensures it has the required specifications and sequence. However, this method performs successfully only when the Bifragment are close to each other (the gap size has the maximum of 80 clusters). Additionally, the decoding and validation conditions do not always lead to correct carving in which the existing of a corrupted cluster causes implies incorrect carving.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

7.3 Restart marker (RSTm)

Karresand and Shahmehri [63] propose Restart Markers RSTm as a carving method of JPEG images. The method has a reassembling technique that uses fragments of non-differential Huffman entropy coded decoder. It uses the start marker and end marker and the restart marker (RSTm) of a JPEG file format to identify and validate the structure and content of the image. The start and end markers are allocated at the beginning and end of a JPEG file while the RSTm sequentially appears at the scan area of the JPEG file. The RSTs index the scan area data pattern to stop the scan at specific intervals. This data is called as Minimum Coding Unit (MCU). Moreover, it builds the Direct Current (DC) coefficient chains of the JPEG image via Discrete Cosine Transform (DCT). The luminance DC values of all restart intervals identify the distance of closest match in order to correctly connect vertically oriented lines of Bifragment. However, the method only handles JPEG images that have RSTm. It does not handle missing RSTm of the RSTm sequence. Furthermore, this technique heavily relies on the image content, which entails exhaustive identification and validation process.

7.4 Sequential Pixel Prediction

Guo and Xu [47] presented a Sequential Pixel Prediction (SPP) content-based method to recover fragmented JPEG file. The SPP method uses a neural network as a reassembling technique to predict the sequence of clusters (the current fragment is adjacent to the following fragment). The neural network assigns weights to the cluster and manipulates the weights according to measurements of the image width of data and backpropagation error analysis to reassemble fragment JPEG file. The advantage of this technique is that it only relies on the actual contents of the image fragments. The disadvantage is that this technique successfully performs the reassembly when only a few clusters are not in sequential order. Additionally, it does not handle fragmented clusters of another file, hence it does not recover images that include intertwined images such as the thumbnails.

7.5 Multimedia Files Carving

Qiu et al. [55] propose a new Multimedia File Carving (MFC) method using Parallel Unique Path (PUP) reassembling technique. The aim of the MFC method is handling high entropy file fragments with high recovery accuracy. The carving process includes three main steps: (i) identifying header of a file, (ii) classifying the file fragment types, and (iii) reassembling the file. The first step is performed based on entropy, BFD and RoC data, the second step is performed by SVM to classify the entropy, BFD and RoC features and reassembling the file fragments. The entropy, BFD, and RoC are signature and statistical features. The MFC method is meant to handle complex and disordered (non-linear) fragments. However, it is found to be inaccurate and inefficient in handling regular recovery cases. Additionally, it does not handle fragmented clusters of another file, hence it does not recover images that include intertwined images such as the thumbnails.

7.6 X_myKarve

Abdullah [53] propose a X_myKarve method to carve fragmented JPEG image. The X_myKarve is based on the myKarve method [13]. The myKarve identifies file structure of fragmented JPEG images using a Define Huffman Table (DHT). It validates the content of intertwined images by checking headers and selecting markers of the images. The reassembly operation is simply reconstructing the images when they are found to have valid contents. Both myKarve and X_myKarve handle contiguous fragmented and linearly JPEG images along with their thumbnails.

Additionally, both methods handle fragmented files, Bifragmented, and intertwined fragmented cases but they have a different technique in performing the reassembly process in handling them. myKarve only deletes the fragmented or unrelated clusters while the X_myKarve use a binary search for a fragmentation point detection then reconstruct the possible options of the image file. However, both methods do not work on the cases that entail processing on the scan area of image files. Subsequently, the methods do not consider identifying and validating other useful information related to the scan area which might improve the reassembling results.

7.7 JPGcarve

De Bock and De Smet [16] present a comprehensive JPGcarve method that recovers different cases of non-fragmented and fragmented JPEG image files. The JPGcarve method performs recovery according to four main steps, which are identification cluster size, and offset via estimation, search space reduction for fragment matching,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

carving single-fragment JPEG image, and carving multi-fragment JPEG image. The objective of the first step is to identify the fragmentation point by sequentially comparing adjacent pairs of clusters from the starting point until the fragmentation point. The second step aims to validate the image content and reduce the carving errors. The third and fourth steps reconstruct the image based on its fragmentation case. However, a simple sorting algorithm without deep analysis of the image content performs the reassembling process. This issue affects the accuracy of the recovery in general as many image recovery cases have complex recovery scenarios.

8. ANALYSIS AND DISCUSSION

The literature review of digital forensic outcomes traditional file recovery methods that use markers like headers and footers to identify and reconstruct the JPEG parts of a file. The current advance studies in digital forensic focus on improving carving methods accuracy in recovering files that having missing file systems or exposed to fragmentation. Mostly, these studies focus on carving JPEG file format. There are three types of carving methods: Signature-based, Structure-based, and content-based carvers. The carving operation includes three main steps of identification, validation and reassembling.

The identification step of digital fragments contributes to the file reassembly or recovery process by both recognizing fragments that are likely relevant or irrelevant for subsequent analysis. The validation step includes a list of structural and computational requirements of a file that are needed for its recovery. The validation limit depends on the number of digital fragments that need to be evaluated. The validation improves the efficiency, accuracy, and robustness of the recovery method. Finally, the reassembling step attempts to reconstruct the file based on available data. Appendix A summarizes the main points of interest of the related work. It contains the proposed methods, their recovery steps, the applied techniques or algorithms in each step, the used criteria to evaluate the performance of the methods, and the remarks of the methods.

Subsequently, the analysis of the proposed methods of the related work shows the various recovery conditions of JPEG files. The previous work does not consider reassembling JPEG file intertwined with non-JPEG file and/or Bifragmented file with/out RSTm in scan area. Most of the cases that are undertaken in the scan area lack deep analysis of the contents and comprehensive view to possible reconstruction. Additionally, handling multi-fragment cases are an extremely complex issue that is yet to be conveniently solved. The analysis of the related work also reveals that there exist a few attempts to apply Artificial Intelligence and machine learning techniques in the file recovery.

There is a wide range of artificial intelligence applications in the literature. Some examples are [66], [67], [68] and [69]. The Artificial Intelligence techniques are found to be not fully utilized in this field. The main usage of the techniques is in the identification step in which it classifies the clusters based on a number of extracted features. Examples are Neural Network, SVM, and genetic algorithm. Some Artificial Intelligence techniques are also applied in the reassembling part as in the work of Xu and Dong [47] in which a neural network operates on the scan area to reconstruct JPEG files. Table 1 presents the application of a number of Artificial Intelligence techniques in file recovery.

Table 1: The Artificial Intelligence in file recovery

Technique	Usage
Neural Network	Predict next fragment to recover JPEG file [47].
	Applied to a classifier for a type detection [61].
Genetic algorithm	Predict types of file fragments [30].
	Improving the identification of file clusters [27].
Support Vector Machine	Improve a recovery accuracy of high entropy file fragments [55].
	Identifying fragment types [71].
	Clusters classification [27].
Extreme Learning Machine	Recovering images file [59].
	Clusters classification [74].
Kohonen Neural Networks	Automatic color based 2-D image fragment reassembly [56].
k-nearest neighbor	Distance metric between file fragments [72].
Software agent	Perform a number of files recovery tasks [5].
Expert system	Assists non-expert users in files recovery [73].

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

9. CONCLUSIONS AND REMARKS

Digital forensics implies applying file recovery methods on bit-copy images of a disk drive to extract potential evidence. The recovery entails preserving different pieces of allocated and unallocated data for investigation. This paper adopts the carving methods as a base work due to their comprehensive view for the recovery conditions of JPEG files. Furthermore, this paper views the different cases and causes of JPEG file fragmentation. The less covered cases in the literature are JPEG file intertwined with non-JPEG file and/or Bifragmented file with/out RSTm and multi-fragmented JPEG file and all of which in the scan area. The conventional carving methods could not handle or are not very effective in handling these cases in which files are partially overwritten or heavily fragmented. Additionally, most of the cases that are undertaken in the scan area lack deep analysis to the contents and comprehensive view to the possible reconstruction. Subsequently, this work recommends to study and deploy Artificial Intelligence techniques and algorithms in JPEG file carving and recovery.

ACKNOWLEDGMENT

This work is supported by the faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia under Grant Vote No. U495.

REFRENCES:

[1] J. Cai, L. Dawson, G. T. Javan, S. Özsoy, F. C. Quaak, and T. K. Ralebitso-Senior, “From Experimental Work to Real Crime Scenes and the Courts”, In Forensic Ecogenomics, pp. 177-209, 2018.

[2] N. Alherbawi, Z. Shukur, & R. Sulaiman, (2013). Systematic literature review on data carving in digital forensic. Procedia Technology, 11, 86-92.

[3] R. K. Pahade, B. Singh, and U. Singh, “A Survey on Multimedia File Carving”, International Journal of Computer Science & Engineering Survey, Vol.6, No.6. 2015.

[4] J. Zhang, and Q. Dong, “Efficient ID-based public auditing for the outsourced data in cloud storage”. Information Sciences, 343, 1-14, 2016.

[5] V. Ganesh, “Artificial Intelligence Applied to Computer Forensics”, International Journal, 5(5), 2017.

[6] A. Sorokin, and E. Makushenko, “Identification of JPEG files fragments on digital media using binary patterns based on Huffman code table”, In Digital Information Processing, Data Mining, and Wireless Communications, pp. 137-141, 2016.

[7] A. Singh, N. Jindal, and K. Singh, “A review on digital image forensics.” International Conference on Signal Processing, page 12-6, 2016.

[8] K. Kendall, J. Kornblum, and N. Mikus, “Foremost - latest version 1.5.7,”, 2010 <http://foremost.sourceforge.net/, 2010, accessed 26-03-18>.

[9] G. G. Richard, and V. Roussev, “Scalpel: A Frugal, High Performance File Carver", Proceeding of Digital Forensics Research Workshop, 2005.

[10] J. Metz, and R. J. Mora, “Analysis of 2006 DFRWS Forensic Carving”. DFRWS (2006) Challenge. <http://sandbox.dfrws.org/2006/mora/dfrws2006.pdf>.

[11] M. I. Cohen, “Advanced Carving Techniques. Digital Investigation”, 4(1-4), pp. 119-128, 2007.

[12] R. Poisel, “Multimedia file carving”, 2012.

[13] K. M. Mohamad, A. Patel, T. Herawan, and M. M. Deris, “myKarve: JPEG image and thumbnail carver”, Journal of Digital Forensic Practice, 3(2-4), 74-97, 2010.

[14] N. Memon, and A. Pal, “Automated reassembly of the file fragmented images using greedy algorithms”. Journal IEEE Trans, 15(2), pp. 385-393, 2006.

[15] A. Pal, H. T. Sencar, and N. Memon, “Detecting File Fragmentation Point Using Sequential Hypothesis Testing”, Digital Investigation, 5, pp. S2-S13, 2008.

[16] J. De Bock, and P. De Smet, “JPGcarve: an advanced tool for automated recovery of fragmented JPEG files”, IEEE Transactions on Information Forensics & Security, 11-1, 19-34., 2016.

[17] N. Alherbawi, Z. Shukur, and R. Sulaiman, “Systematic literature review on data carving in digital forensic”, Procedia Technology, 11, 86-92, 2013.

[18] R. Poisel, and S. Tjoa, “A comprehensive literature review of file carving”, In Availability, reliability and security, eighth international conference on, pp. 475-484, 2013.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[19] D. Povar, and V. K. Bhadran, "Forensic data carving", In International Conference on Digital Forensics and Cyber Crime, pp. 137-148, 2011.

[20] S. H. Khaleefah, M. F. Nasrudin, and S. A. Mostafa, "Fingerprinting of deformed paper images acquired by scanners", In Research and Development, IEEE Student Conference on, pp. 393-397, 2015.

[21] D. Quick, and K. K. R. Choo, "Data reduction and data mining framework for digital forensic evidence", storage, intelligence, review and archive, 2014.

[22] S. L. Garfinkel. Digital forensics research, "The next 10 years digital investigation", 7, S64-S73, 2010.

[23] E. Durmus, M. Mohanty, S. Taspinar, E. Uzun, and N. Memon, "Image carving with missing headers and missing fragments", In Information Forensics and Security, pp. 1-6, 2017.

[24] W. J. Bodziak, 'Forensic footwear evidence", CRC Press, 2017.

[25] S. L. Garfinkel, "Carving Contiguous and Fragmented Files with Fast Object Validation", Digital Investigation, 4-1, pp. S2-S12, 2007.

[26] Y. Tang, J. Fang, K. P. Chow, S. M. Yiu, J. Xu, B. Feng, and Q. Han, "Recovery of heavily fragmented JPEG files", Digital Investigation, 18, S108-S117,2016

[27] B. Roux, "Reconstructing Textual File Fragments Using Unsupervised Machine Learning Technique", 2008.

[28] A. Dewald, M. Luft, and J. Suleder, "Incident Analysis and Forensics in Docker Environments", 2018.

[29] G. Richard, V. Roussev, and L. Marziale, "In-place file carving". In International Conference on Digital Forensics, pp. 217-230, 2007.

[30] W. C. Calhoun, and D. Coles, "Predicting the types of file fragments", digital investigation, 5, S14-S20, 2008.

[31] N. A. Abdullah, R. Ibrahim, and K. M. Mohamad. "Cluster size determination using JPEG files". In International Conference on Computational Science and Its Applications, pp. 353-363, 2012.

[32] M. Nadeem Ashraf, "Forensic Multimedia File Carving", Master's Thesis, KTH, 2013.

[33] A. Pal, and N. Memon, "The evolution of file carving", IEEE Signal Processing Magazine, 26(2). pp. 59-71, 2009.

[34] A. B. Lewis, "Reconstructing compressed photo and video data", Doctoral dissertation, University of Cambridge, No. UCAM-CL-TR-813, 2012.

[35] N. A. Abdullah, "X_myKarve: Non-Contiguous JPEG File Carver", International Journal of Digital Crime and Forensics, 8-3, 63-84, 2016.

[36] B. Carrier, "File system forensic analysis", Addison-Wesley Professional, 2005.

[37] B. Birmingham, R. A. Farrugia, and M. Vella, "Using thumbnail affinity for fragmentation point detection of JPEG files", In Smart Technologies, International Conference on, pp. 3-8, 2017.

[38] S. L. Garfinkel, "File Carving". 2012.

[39] CCITT (1992), The International Telegraph & Telephone Consultative Committee (CCITT). Information Technology Digital Compression and Coding of Continuous Tone Still Image Requirements and Guideline. T.81. 1992.

[40] G. S. V. R. K. Rao, P. Jinka, V. Srinivasan, R. Selvaraj, S. K. Ramaswamy, and D. Maroo, "System and method for automatically extracting multi-format data from documents and converting into XML", 2015.

[41] J. D. Huang, "The JPEG Standard", Graduate Institute of Communication Engineering National Taiwan University, 2006.

[42] C. Nguyen, "Computer Faults in JPEG Compression and Decompression Systems". Electrical and Computer Engineering University of California, Davis Davis, CA 95616, 2002.

[43] C. E. Shannon, "A mathematical theory of communication", Bell System Technical Journal, 27, pp. 379-423 and 623-656, 2001.

[44] K. M. Mohamad, and M. M. Deris, "Visualization of JPEG metadata", In International Visual Informatics Conference, 2009.

[45] H. Guo, and M. Xu, "A method for recovering jpeg files based on thumbnail", Proceeding of the International Conference Automation and Systems Engineering, 1-4, pp. 2011.

[46] M. Xu, J. Sun, N. Zheng, T. Qiao, Y. Wu, K. Shi, and T. Yang, "A Novel File Carving Algorithm for EVTX Logs", In International Conference on Digital Forensics and Cyber Crime, pp. 97-105, 2017.

[47] M. Xu, and S. Dong, "Reassembling the fragmented JPEG images based on sequential pixel prediction", In Computer Network and Multimedia Technology, International Symposium on, pp. 1-6, 2009.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[48] K. Shi, M. Xu, H. Jin, T. Qiao, X. Yang, N. Zheng, and K. K. R. Choo, "A novel file carving algorithm for National Marine Electronics Association logs in GPS forensics", Digital Investigation, 23, 11-21., 2017.

[49] E. Uzun, and H. T. Sencar, "Carving orphaned JPEG file fragments", IEEE Transactions on Information Forensics and Security, 10(8),1549-1563, 2015.

[50] S. J. J. Kloet, "Measuring and improving the quality of file carving methods", Master's Thesis, Niederlande: Eindhoven University of Technology 4-79, 2007.

[51] DFRWS 2006, Digital Forensics Research Conf. (DFRWS), "DFRWS 2006 Forensics Challenge," <http://www.dfrws.org/2006/challenge/, 2006, Retrieved July 7th 2016>.

[52] DFRWS 2007, Digital Forensics Research Conf. (DFRWS), "DFRWS 2007 Forensics Challenge," <http://www. dfrws.org/2007/challenge/, 2007, Retrieved July 7th 2016.>\

[53] N. A. Abdullah, "An improved file carver of intertwined jpeg images using X_myKarve", Doctoral dissertation, Universiti Tun Hussein Onn Malaysia, 2014.

[54] H. T. Sencar, and N. Memon, "Identification and Recovery of JPEG Files with Missing Fragments", Digital Investigation, 6, pp. S88-S98, 2009.

[55] W. Qiu, R. Zhu, J. Guo, X. Tang, B. Liu, and Z. Huang, "A new approach to multimedia files carving", In Bioinformatics and Bioengineering, International Conference on, pp. 105-110, 2014.

[56] E., Tsamoura, & I. Pitas, "Automatic color based reassembly of fragmented images and paintings", IEEE Transactions on Image Processing, 19(3), 680-690, 2010.

[57] M. C. Stamm, S. K. Tjoa, W. S. Lin, and K. R. Liu, "Anti-forensics of JPEG compression", In Acoustics Speech and Signal Processing, 2010 IEEE International Conference on, pp. 1694-1697, 2010.

[58] Q. Li, A. Ong, P. Suganthan, and V. Thing, "A novel support vector machine approach to high entropy data fragment classification", In Proc. South African Information Security Multi-Conf., pp. 236-247, 2011.

[59] L. Zhang, D. Zhang, and F. Tian, "SVM and ELM: Who Wins? Object recognition with deep convolutional features from ImageNet", In Proceedings of ELM-2015 Volume 1, pp. 249-263, 2016.

[60] N. Mehra, and S. Gupta, "Survey on multiclass classification methods", International Journal of Computer Science and Information Technologies, vol. 4 (4), 572 – 576, 2013.

[61] M. C. Amirani, M. Toorani, and S. Mihandoost, "Feature-based type identification of file fragments", Security and Communication Networks, 6(1), 115-128, 2013.

[62] M. McDaniel, and M. H. Heydari, "Content based file type detection algorithms", In System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on, pp. 10-pp, 2003.

[63] M. Karresand, and N. Shahmehri, "Oscar-file type identification of binary data in disk clusters and ram pages". Security and privacy in dynamic environments, 413-424, 2006.

[64] C. J. Veenman, "Statistical disk cluster classification for file carving", In Information Assurance and Security, 2007. IAS 2007. Third International Symposium on, pp. 393-398, 2007.

[65] A. Pal, K. Shanmugasundram, and N. Memon, "Automated Reassembly of Fragmented Images", Proceeding of the 2003 Acoustics Speech and Signal Processing, 2003.

[66] M. A. Mohammed, M. K. A. Ghani, R. I. Hamed, S. A. Mostafa, D. A. Ibrahim, H. K. Jameel, and A. H Alallah, "Solving vehicle routing problem by using improved K-nearest neighbor algorithm for best solution", Journal of Computational Science, 21, 232-240, 2017.

[67] M. A. Mohammed, B. Al-Khateeb, A. N. Rashid, D. A. Ibrahim, M. K. A. Ghani, and S. A. Mostafa, "Neural network and multi-fractal dimension features for breast cancer classification from ultrasound images", Computers and Electrical Engineering, 2018.

[68] S. A. Mostafa, A. Mustapha, M. A. Mohammed, M. S. Ahmad, and M. A. Mahmoud, "A fuzzy logic control in adjustable autonomy of a multi-agent system for an automated elderly movement monitoring application", International journal of medical informatics, 112, 173-184, 2018.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[69] S. A. Mostafa, A. Mustapha, S. H. Khaleefah, M. S. Ahmad, and M. A. Mohammed, "Evaluating the Performance of Three Classification Methods in Diagnosis of Parkinson's Disease", In Recent Advances on Soft Computing and Data Mining, Springer, Cham, pp. 43-52, 2018.

[70] L. Aronson, and J. Van Den Bos, "Towards an engineering approach to file carver construction", In Computer Software and Applications Conference Workshops, 2011 IEEE 35th Annual, pp. 368-373, 2011.

[71] L. Sportiello, and S. Zanero, "File block classification by support vector machines", in Proc. of the 6th Int. Conf. on Availability, Reliability and Security, pp. 307–312, 2011.

[72] S. Axelsson "The normalized compression distance as a file fragment classifier", In: Proceedngs of the 2010 Digital Forensics Research Conference (DFRWS); 2010.

[73] U. Karabiyik. "Building an intelligent assistant for digital forensics". Doctoral dissertation, The Florida State University, 2015.

[74] R. R. Ali, K. M. Mohamad, S. Jamel, & S. K. A. Khalid, (2018, February). Classification of JPEG Files by Using Extreme Learning Machine. In Recent Advances on Soft Computing and Data Mining, Springer, Cham, pp. 33-42 pp. 2018.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

APPENDIX A: THE SUMMARY OF THE CARVING METHODS

Section	Steps	Technique/Algorithm	Evaluation criteria	Remarks
7.1	Carving	Structure-based of JPEG file	Accuracy of recovered image or thumbnail	• DFRWS (2006-2007) dataset • Linear-fragmented JPEG file with a complete file type
	Identification	Standard/Non-standard headers		
	Validation	Extension file type		
	Reassembling	Not exist		
7.2	Carving	Content-based of JPEG file	Accuracy of recovered fragmented image	• DFRWS (2006) dataset • Linear-Bifragmented JPEG file caused by a gap
	Identification	Standard headers marker		
	Validation	Object validation		
	Reassembling	Header/footer marker		
7.3	Carving	Content-based of JPEG file	Fragment joint validity value	• Not specified dataset • Linear-fragmented JPEG file that has RSTm in the scan area
	Identification	Standard header markers		
	Validation	Rate of Change (RoC)		
	Reassembling	DCT technique in a scan area		
7.4	Carving	Content-based of JPEG file	Accuracy of recovered fragmented image	• DFRWS (2006) dataset • Linear-Bifragmented JPEG file in internal contents
	Identification	Standard headers marker		
	Validation	Image structure		
	Reassembling	Neural network classification		
7.5	Carving	Content-based of JPEG file	Accuracy of recovered image or thumbnail	• DFRWS (2006-2007) dataset • Linear and non-linear-fragmented JPEG file
	Identification	Standard headers marker		
	Validation	SVM classification		
	Reassembling	Parallel Unique Path (PUP)		
7.6	Carving	Signature of JPEG file	Accuracy of recovered image or thumbnail	• DFRWS (2006-2007) dataset • Linear-fragmented JPEG file caused by garbage
	Identification	Standard/Non-standard headers		
	Validation	DHT Patterns		
	Reassembling	Matching algorithm		
7.7	Carving	Content-based of JPEG file	Accuracy of recovered image or thumbnail	• DFRWS (2006-2007) dataset • Linear-intertwined JPEG file
	Identification	Standard/Non-standard headers		
	Validation	UHP Patterns		
	Reassembling	Binary search algorithm		
7.8	Carving	Content-based of JPEG file	Accuracy and time cost of recovered image	• DFRWS (2006-2007) dataset • Non-fragmented and Multi-fragmented JPEG file
	Identification	Standard header markers		
	Validation	Matching algorithm		
	Reassembling	Sorting algorithm		