


DEJAVU FORENSICS: ENHANCING RECOVERY OF FORMATTED JPEG AND PNG DATA USING SUPPORT VECTOR MACHINES

 <https://doi.org/10.56238/arev7n3-301>

Date of submission: 28/02/2025

Date of publication: 28/03/2025

Islan Amorim Bezerra¹, Rubens Karman Paula da Silva², Sidney Marlon Lopes de Lima³, Sergio Murilo Maciel Fernandes⁴, Carolina de Lira Matos⁵ and Jheklos Gomes da Silva⁶

ABSTRACT

With technological advancements, virtual crimes are occurring more frequently. When digital equipment is stolen, lost, or discarded, the data remains stored on the disks, enabling its recovery. This work focuses on the recovery of formatted files, investigating the applicability of the tools Foremost, Scalpel, and Magic Rescue in Linux, as well as an in-house tool equipped with machine learning. The goal is to develop a tool for the recovery and validation of formatted files, contributing to investigations of digital crimes and bringing new insights into recovery methods. Using pattern recognition, the cluster is used as input, acting as a neuron in the learning machine. The work applies machine learning to recognize patterns in blocks/clusters. In the "simple" scenario, the classification is binary (class vs. counter class), a methodology developed by Pavel (2017). In the "complex" scenario, the one-against-all method was used, with a database of 16,000 files. The research presents

¹ PhD student in Computer Engineering
University of Pernambuco – UPE
E-mail: iab@ecomp.poli.br
ORCID: <https://orcid.org/0000-0002-8920-6691>
Lattes: <http://lattes.cnpq.br/0162367027095181>

² PhD student in Computer Engineering
University of Pernambuco – UPE
E-mail: rkps@ecomp.poli.br
ORCID: <https://orcid.org/0000-0001-9388-9889>
Lattes: <http://lattes.cnpq.br/6372811203248481>

³ Post-Doctorate in Computer Engineering
University of Pernambuco – UPE
E-mail: smll@ecomp.poli.br
ORCID: <https://orcid.org/0000-0002-4350-9689>
Lattes: <http://lattes.cnpq.br/0323190806293435>

⁴ PhD in Computer Engineering
University of Pernambuco – UPE
E-mail: smurilo@ecomp.poli.br
ORCID: <https://orcid.org/0000-0002-5922-4119>
Lattes: <http://lattes.cnpq.br/4520293519781462>

⁵ Undergraduate student in Electrical and Electronic Engineering
Federal University of Pernambuco – UFPE
E-mail: carolina.liram@ufpe.br
ORCID: <https://orcid.org/0009-0009-2069-869X>
Lattes: <http://lattes.cnpq.br/8418415081574622>

⁶ PhD student in Computer Engineering
University of Pernambuco – UPE
E-mail: jheklos.gomess@upe.br
ORCID: <https://orcid.org/0000-0002-7741-8849>
Lattes: <http://lattes.cnpq.br/3048498549751735>

an approach that combines machine learning and data science to recover formatted data. The in-house tool achieves a recovery rate of over 96% for formatted PNG and JPEG files, running in seconds.

Keywords: Data carving. Digital forensics. Cybercrime. Data recovery. Machine learning.

INTRODUCTION

In the digital age, access to information is driving significant advances in many areas of knowledge. These advances contribute to the development of tools to optimize processes. On the other hand, the rise and constant evolution of technology have created an environment that is conducive to the emergence and spread of digital crime. These cases are characterized by illegal activities that compromise the integrity, confidentiality, and security of users.

Digital forensics is responsible for the clarification of digital crimes based on the search for evidence and facts that incriminate the suspects and protect the victims. The search for evidence can be carried out on devices such as smartphones and computer networks. The focus is on law enforcement, so that the investigation is properly conducted and constitutes admissible evidence in a court of law.

The relationship between digital crime and data recovery is given by user usage. Today's users use their devices, such as smartphones and computers, as a kind of digital archive. Various programs and applications are downloaded for a variety of functions. The files are saved for later use, and those that shouldn't be saved are supposed to be deleted.

Metadata scanning is vital in digital crime investigations. It provides detailed information about files, listing their creation, modification, and access dates, device type, and permissions. This data helps forensic experts investigate suspicious activity, such as unauthorized access and document tampering, and also helps them reconstruct timelines. However, sophisticated cybercriminals can hinder this analysis by deleting or overwriting metadata through device formatting. This makes it harder to recover critical information and complicates investigators' work in tracking down digital evidence.

File carving is a data recovery technique. It is a tool in digital forensics and serves as a final option when other recovery methods are no longer available. This approach becomes essential in cases where it is impossible to reconstruct file system metadata, such as with formatted hard disks.

File carving extracts file fragments without using file system structures. This process analyzes the raw data on the disk, identifying signatures or patterns that are unique to different types of files. This feature allows investigators to recover files from a formatted drive.

When used improperly, file carving can become a powerful tool in the hands of cybercriminals. If an electronic device is lost or stolen, the data often remains on the

storage. Malicious actors can recover it even after deletion or formatting. Fragments of these files can be recovered, which may expose sensitive data. The main risk lies in the recovered information. Misuse of restored confidential files, personal images, or bank details can occur, facilitating fraud and other crimes.

The primary objective of this article is to explore the application of file carving techniques in digital forensics, focusing on the recovery of formatted data in scenarios involving digital crimes. By analyzing the effectiveness of tools such as Foremost, Scalpel, and Magic Rescue, as well as introducing an innovative machine learning-based tool, Dejavu Forensics, this study aims to enhance the recovery process of formatted files, particularly JPEG and PNG formats. Additionally, the research seeks to demonstrate how machine learning, specifically Support Vector Machines (SVM), can improve data recovery accuracy and efficiency. Ultimately, this work contributes to the field of digital forensics by providing insights into the challenges of recovering formatted data and offering solutions that can aid in the investigation of digital crimes.

THEORETICAL FRAMEWORK

THE RELATIONSHIP BETWEEN DIGITAL CRIME AND FORMATTED DATA

Digital crime includes various malicious acts that compromise computer systems, data, and online operations. The rise of cyber-attacks shows we must understand their impact on digital crime and data recovery. Digital crime is a complex issue that includes exploiting security flaws, using malware, and conducting attacks in a digital environment. It also involves actions that compromise user data. The rise in digital attacks demonstrates criminals' evolving tactics in online spaces.

Another critical situation with high incidence rates is sextortion. According to the Cybercrime Observatory, sextortion refers to sexual extortion, such as intimate videos or photos. Criminals gain access to their victims' intimate content and coerce them with the threat of exposure. In most cases, partners or ex-partners commit this crime, but it can also be done by hackers with whom the victim has shared erotic content online (Cybercrime Observatory, 2020). This crime highlights the neglect of the phrase "look and then delete." Even if the file is deleted, the data remains on the device and can be retrieved later. In such cases, digital forensic tools, such as Dejavu Forensics, play a vital role by enabling the recovery of formatted data while ensuring adherence to forensic standards. This ensures the recovered data can be used as evidence in legal investigations, maintaining the integrity

and chain of custody required in forensic procedures.

Recovering formatted data fails if the user overwrites the partition's contents bit by bit, sterilizing the data. This process makes it impractical to recover the original data, and investigators cannot extract any relevant information for the investigation.

As a case study, two graduate students at the Massachusetts Institute of Technology (MIT) Computer Science Laboratory published a report on an experiment where they bought 158 used hard drives, only 129 of which worked. Of these, only 12 were properly "sterilized." For the rest, it was possible to recover more than 5,000 credit card numbers, email folders, bank transactions, email addresses, and hospital records (Garfinkel, 2003).

However, as illustrated in the experiment conducted by the two MIT students, many users, intentionally or not, do not apply data sterilization properly. The investigation found thousands of sensitive personal details on supposedly deleted hard drives (Garfinkel, 2003).

There are various tools for sterilizing data, but they are not native to Windows or smartphone operating systems. On Android, users can perform light formatting by pressing two buttons for 10 seconds, which is easy. But sterilization is more complex, as there are no native methodologies available. Normally, users need to connect the smartphone to a desktop computer, where line commands must be used for sterilization. Furthermore, this procedure has drawbacks, as it can void the warranty, and Android prevents access to external operating systems.

In this sense, formatted data, due to its structured and critical nature, is one of the most targeted links in illicit activities in the digital environment through various invasion and exploitation techniques. The possibility of recovering formatted data becomes a target because it is an environment rich in sensitive information, and improper access or manipulation can have negative consequences.

Therefore, the investigation of digital crimes becomes the responsibility of computer forensics, which combines elements of law and computer science to obtain traces and evidence of digital crimes (Casey, 2011). Digital forensics is a recent activity, with its first practices emerging in the 1980s when the first cases of viruses in communication networks appeared. In the following years, investigations evolved into cases of pedophilia and later into broader cybercrime (Hannan, 2004).

In forensics, data recovery is the process of restoring lost files, which helps computer professionals gain new insights into the analysis of recovery methods for formatted files.

This method is defined as the process of recovering deleted or logically formatted data (Vacca, 2006).

OVERVIEW OF FORMATTED DATA RECOVERY

Recovering formatted data is possible. When a user deletes a file, it is logically deleted from the storage system. The operating system (OS) knows which parts of the device the file occupies. In simple terms, the OS only modifies the space used by that file, making it available for use. However, the data remains on the device. Therefore, it is possible to recover the data (Pal, 2009).

The allocation table is updated to indicate that the blocks previously assigned to the removed file are free. However, the logically formatted data remains on the device. The data stays until, for example, a new file replaces it or when an existing file is expanded.

Even if a user deletes the data using the delete command, the data may remain on the device. The same happens with the format command, which is used to logically prepare the disk for use but does not guarantee that the data will be destroyed (James, 2006).

Digital forensics techniques use residual information as a resource for recovering files, such as fragments that the user thought had been deleted. As such, data recovery can act as potential evidence of digital crime. It should be emphasized that these techniques are used for both positive and negative purposes.

DATA CARVING AND ITS LIMITATIONS

Carving is a general term for extracting raw data files from a file system. It is based on the specific characteristics of the file format (Alherbawi, 2013). Data carving, in computing, refers to the recovery of deleted files. However, it corresponds to a technique performed by locating known signatures.

When a disk is logically formatted, a table is created that acts as a kind of map. This map guides the read and write heads to the correct positions where each recorded file is located. This table contains the file system, which determines: (i) how the files will be accessed (sectors) and (ii) the size of the clusters.

Computer storage is organized into units named sectors. The file system, in turn, groups these sectors into smaller units, known as blocks/clusters (Ali, 2018). Clusters are characterized by having headers that work as a marker for the beginning of the files. This signature describes the file system methods as well as the type of file and content.

These signatures consist of a hexadecimal header and footer, and each file type has a unique identifier. A header identifies the bytes at the beginning of the file, while the footer identifies the bytes at the end of the file.

In summary, for each type of file or category of files, a different technique is required. Through data carving, it is necessary to check the structure of the clusters to decide if they are consistent and whether they can still be considered a coherent unit of the file system (Sencar, 2009). Carrier (2005) describes a file system as structural data and user organization in a way that the machine can find it.

Data carving techniques are most often used to recover files that are not allocated on the drive. Allocated space, in this sense, refers to a partition on the drive. This partition does not show any file information, as if they were damaged or missing (Pal, 2009). The data carving approach was pioneered by the Defense Computer Forensics Lab (DCFL) producing "Carv This." Next, Kriss Kendall and Jesse Kornblumt presented 'Foremost,' which corresponds to a code tool (Sari, 2020).

According to Kent, the forensic process comprises four phases, as described in Figure 1 (Kent, 2006). The collection phase aims to label, identify, and acquire the data. The examination phase involves automating processes and combining data in order to extract the relevant data. Analysis uses methods and techniques to acquire information based on collected data. Finally, the results obtained correspond to the final reports of the analysis, which include the procedures used and possible improvements that can be made.

Figure 1 - Diagram of forensic process



Source: Prepared by the authors themselves

One of the main features of this technique is the signature of the file structure. PDF extension files, for example, have the initial signature, the header. That is, files in PDF

format always start the same way, with the same header. JPEG files, in turn, have in their structure the hex header “0xFFD8” and the footer “0xFFD9”, as shown in Figure 2. This makes it possible to distinguish it from other types of files, based on an examination of the content. The header and footer sequence search is also known as "magic numbers".

File systems are responsible for structure management. Furthermore, they allocate clusters that are either sequential or not. The absence of a string in the clusters indicates a storage problem known as fragmentation. These fragments can be distributed in any order, depending on the system, file sizes, and cluster size.

Despite the benefits that Data Carving can provide, there are some significant limitations. Such limitations may interfere with ensuring the effectiveness of recovery of formatted data. Being: i. the generation of false-positive files, which are usually corrupted files; ii. the possibility of discarding files that, in fact, exist in the system; iii. in addition to disregarding possible fragments.

Data Carving Limitations: False Positives

During a criminal investigation process it is necessary to consider the use of methods and protocols. The goal is to ensure effectiveness in the data being retrieved. It is necessary to ensure effectiveness during this process. It is important that the tool used generates as few false-positive files as possible.

When a digital data recovery process is performed, real files are generated. However, files that are incorrect or corrupted can be generated, named false positives (Laurenson, 2013). False positives are files that never existed and that can be generated in large numbers.

Because they do not exist, these fake files can be very large. Their presence tends to be synonymous with difficulty in the process carried out by the criminal expert.

The greater the number of false-positives, the greater the difficulty in separating real files from those that do not exist.

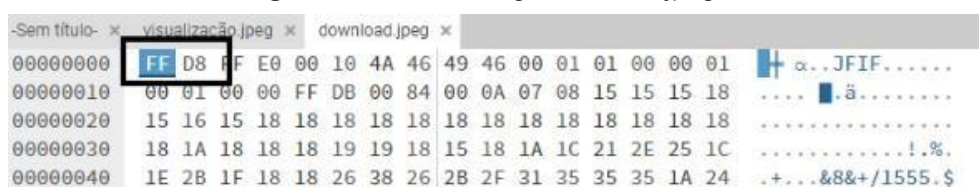
One of the reasons for the large number of false positives is the coincidences between the cluster headers. For example, consider a certain cluster containing the initial characters “\%PDF”. In this cluster, there would be recovery of a previously deleted pdf file. But it may be a simple coincidence that there is a string equal to the header at the beginning of the probed memory device cluster.

Data Carving would start retrieving a pdf file that, in fact, never existed. As the file

does not exist, it may be that its footer is not present in any of the following clusters. As a consequence, the fake file can have gigantic sizes, for example, the fake file could have the size of the partition.

Data Carving determines stopping criteria when searching for the footer in the configuration files. It acts as a palliative measure. The Scalpel tool, based on Data Carving, from its configuration eliminates unwanted metadata in the first phase. It warns that there are some headers and footers that can generate a lot of false positive files. In addition to determining minimum and maximum sizes that files can contain.

Figure 2 - Data Carving: Header of jpeg file



Source: Prepared by the authors

Figure 3 - Data Carving: Footer of jpeg file.



Source: Prepared by the authors.

Data Carving Limitations: Existing Files Ignored

File systems have metadata describing their respective structures. Each structure has a specific sequence in headers and footers. This determines the file ID.

During header signature identification, files can be dropped. Data Carving starts the recovery process. This process is only terminated when finding the footer of the file. However, the header ends up being ignored. Because Data Carving is looking for the end of the corresponding file; the footer.

In this context, DECA – Decision-Theoretic Carving - emerges as a tool capable of reducing the number of false positives (Gladyshev, 2017). It happens through machine learning techniques - specifically SMV. The first step consists of fetching the header at the beginning of the cluster. In the second step, machine learning is performed. That occurs between the analyzed cluster and its respective type of file through the use of pattern recognition.

Impact of Block/Cluster Size on Data Recovery

The size of the blocks or clusters in file systems plays an important role in the data recovery process. Smaller clusters allow for more detailed data recovery, as each block contains a smaller amount of data. However, this can increase the time required for recovery, as more clusters need to be processed.

On the other hand, larger clusters speed up the read and recovery process, but can increase the possibility of data loss, since more information is stored in each block. This can impact recovery accuracy, especially when there is data corruption in a specific block.

Studies suggest that cluster size affects data recovery tools. It impacts both the time and accuracy of the process (Alherbawi, 2013). Tools that deal with larger clusters tend to be faster, but may face challenges in terms of the integrity of the recovered data.

In our framework, the Sleuth kit tool is used to recover clusters and their associated parameters, such as size and location. We chose the Sleuth kit tool for its robustness and efficiency in forensic analysis. We will also look at its integration with other forensic tools used in the study.

Data Carving Limitations: Cluster Whit Fragments

From a forensic perspective, it is important to be aware of fragmented files. Files can exhibit fragmentation when one or more clusters are not part of the scanned file. Fragmentation usually occurs when data is stored partially. Which can happen due to the free space being reduced or even the deletion of the files.

A fragmentation point is the last cluster in a fragment, that is, where fragmentation occurs. A segment has one or more consecutive fragments that belong to several files (Sari,2020).

Fragmentation occurs when a given file is not stored in the correct sequence in clusters that are consecutive. That is, sequencing the clusters from beginning to end results in incorrect file retrieval. It is as if it provided part, or fragmentation, of what had been stored (Pal; Memon, 2009).

In the study carried out by Garfinkel (2007), the author starts by identifying approximately 350 hard disks. The results indicate that the fragmentation rates of files such as e-mails, JPEG and Microsoft Word tend to be higher. What can happen for several reasons, such as low storage space on the disk, or even wear on the file system.

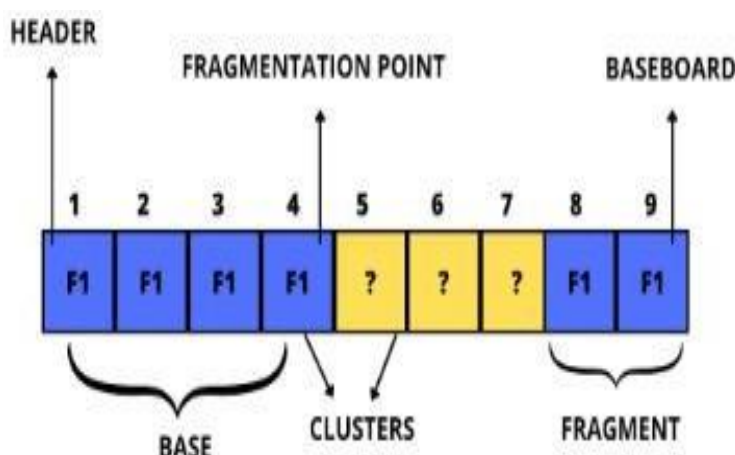
These fragmentations can also be caused by bad blocks. Bad blocks correspond to

corrupted areas on the hard drive itself. Possibly this is due to the defective surface. In this case Data Carving may not discard the bad blocks. Consequently, the recovered file will be corrupted, making it impossible to view and read the file.

In addition to bad blocks, there are still fragments. They are caused by logical destructions that happen improperly by the Operating System. Storage devices use algorithms that perform a type of smoothing, in which the controller reallocates addresses between logical and physical blocks (Pal; Memon, 2009). In general, if the recovered file has fragments, possibly the file is corrupted and cannot be opened.

When the disk has fragmentations, there are some small groups of clusters that are available for storage. What happens is that the possible files that will occupy this space may have larger sizes. As a result these files are stored and during the storage process they are shredded for allocation.

Figure 4 - Data Carving: Bad Blocks..



Source: Adapted from PAL, *et al.* (2009).

In addition to being caused by physical defects, file and cluster corruption can also be caused by undue logical destruction by the Operating System and even by malfunctions after its malfunction after its automatic update. An example of this is the non-detection of fragments using the structure of a bitmap file. In a bitmap file structure, each part of this structure plays an important role as to how the content is displayed to the user, as well as information (metadata) generated and used by the operating system to perform its execution (opening).

The basic organization of the information contained in the .bmp file is:

- File header - file size, initial signature (%bmp) and others;
- Image header - height and width size of the image, compression type, etc;

- Color table - what colors are used by the image;
- Image (content) in pixels - the content itself;

So suppose the recovery tool in use has found an initial bitmap signature (%bmp) and has started Data Carving on cluster $C1$. As mentioned earlier, Data Carving is not able to discard badblocks (or clusters with a fragment), nor can it identify them. So we are faced with the scenario that in a file without a fragment, cluster $C2$ would be responsible for loading the image header information. Whereas in a fragmented file cluster $C3$ will correspond to the image header. So when the .bmp file is opened, it will fetch image height and width information from an invalid cluster. One of the reasons for the large number of corrupted files is that the cluster with fragments contains information necessary for the execution of the file.

USING MACHINE LEARNING TO RECOVER FORMATTED DATA

Algorithms of machine learning are often used to solve pattern recognition problems. The main feature is the ability to generalize in the face of data that were not presented during the training process.

To exemplify some capabilities of AI we can cite: natural language processing, knowledge representation, automated reasoning, computer vision and robotics. Within these capabilities there is a wide range of applications and lines of study. One of them, of interest in this paper, is machine learning for the purpose of formatted data recovery.

Machine learning can intelligently automate many tasks by analyzing thousands of files, extracting features from them, and weighting them statistically. Machine learning can do a lot to improve the security of devices. There are initiatives, but they are still in the early stages (Gladyshev, 2017).

In order to overcome the limitations of Data Carving, the state-of-the-art proposes to extract features of the hard disk cluster preemptively before incorporating it into the file being recovered. It becomes possible to investigate false positives in addition to the presence of bad blocks.

Using the pattern recognition methodology, to extract features from files, the cluster is used as input. It acts as an input attribute of the learning machine. Given this process, the performance of Data Carving may discard clusters that present fragments due to bad blocks. That is, the file analyzed in a pattern recognition process can be compromised by some damage that harmed the hard disk.

Supervised Pattern Recognition

Technological advances allow for large data storage. This makes it possible to find patterns, trends and anomalies that tend to transform data into information, and consequently, information becomes the basis for classifications (Witten *et al.*, 2005). Pattern recognition has the main objective of building a simplified representation of a data set through the characteristics considered most relevant, which enables its partition into classes (Duda; Hart, 2006).

The supervised learning method has as its main characteristic the advanced knowledge of the classes that will be used in the generation of patterns. The disposition, the number of observations, as well as the number of classes are known, based on the measure of similarity of the data in question, when compared to previously labeled data.

Conventionally, there are two methods that can be used to perform classification through supervised learning. The first estimates a probabilistic model based on existing data, in which the estimate is classified using Bayes' theorem. The second method presupposes the discriminant functions that define the so-named decision boundaries that will be used in the classification, based on the data set.

Sensory data through a kind of machine perception. The samples are grouped according to the labels (classes) previously defined by the specialist. The patterns they recognize are numeric, contained in vectors, into which real-world data, whether images, sound, text or time series are translated.

SUPPORT VECTOR MACHINE CLASSIFIER PARAMETERS

"Support Vector Machine" (SVM) is a supervised machine learning algorithm used for problem classification or regression analysis (Mehdi; Amir, 2020). Support Vector Machine is a frontier that best segregates the two classes, using proximity of the data to construct one or more hyperplanes to classify high dimensional data.

The operationalization of a classification task is based on the separation of the data set into "training" and "test". In the attributes that make up the data (features), there is a field named target, which defines the class label. The purpose of SVM is to train models that can predict the target attribute, only taking into account the attributes in the training step (Mehdi; Amir, 2020). If there is no prior expert knowledge about a domain, SVM is an excellent first method to test (Stuart, 2013).

Also, the SVM classification approach can favor forensic analysis and bring more

efficiency to the investigation as it would prioritize the experts' effort. For any variety of case, such as criminal, civil, regulatory, or organizational, where research steps, data extraction, and data analysis are required.

The machine learning methods used by the Support Vector Machine enable application to vectors with large feature sizes, consisting of a statistical learning machine that implements the principles of Structural Risk Minimization - SRM to build the hyperplane. Using this methodology, an input space of non-linearly separable patterns forms a new space in which the dimensions are linearly separable.

The SVM can be defined by an intercept term b and a vector perpendicular to the decision hyperplane, \tilde{w} , known as the weight vector. The training set being T , containing (n) data x_i and its labels y_i , being X the data space and $Y = -1, +1$, the hyperplane is defined in Eq. (1).

$$f(x) = w \cdot x + b = 0 \quad (1)$$

The hyperplane defined by the equation presented separates the vector space of the data into two regions: $\vec{w} \cdot \vec{x} + b \geq 0$ and $\vec{w} \cdot \vec{x} + b \leq 0$. We consider the classification using the signal function.

The figure below illustrates x_1 as being a point on the hyperplane $H1: w \cdot x_1 + b = +1$ and x_2 as a point on the hyperplane $H2: w \cdot x_2 + b = -1$. This projection of $(x_1 - x_2)$ on the weight vector w corresponds to the distance between the hyperplanes presented, and can be calculated from the difference between the hyperplanes, as shown in Eq. (2).

$$\frac{w \cdot x_1 + b = +1}{-w \cdot x_2 + b = -1} \quad (2)$$

$$w \cdot (x_1 - x_2) = 2$$

When the sides of the equality in the equation above are multiplied by $\frac{1}{\|w\|}$, obtaining the value of the distance between the hyperplanes, corresponding to twice the value of the margin as shown in Eq. (3).

$$\frac{w \cdot (x_1 - x_2)}{\|w\|} = \frac{2}{\|w\|} \quad (3)$$

The value obtained is named the geometric margin when taken as its maximum,

which is equivalent to saying that the classifier ρ corresponds to the maximum width of the strip that can be drawn in order to separate the support vectors of the two classes. It is necessary to find w and b where $\rho/\|w\|$ is maximum and for every $(x_i, y_i) \in T, y_i(w_T x_i + b) \geq 1$.

It is common to encounter situations where the data set is not linearly separable. This can occur due to the presence of noise or discrepant data, as well as outliers. In these cases, it is not possible to obtain a hyperplane that separates the classes without classification errors occurring. The strategy is to find the one that produces the least amount of error. The flexible-margin SVM is a method that can handle training sets that have data that may violate the constraints.

The distance from a given x_i to the discriminant is given by Eq. (4).

$$\frac{|w \cdot x_i + b|}{\|w\|} \quad (4)$$

Similarly, when $y_i \in \{-1, +1\}$ it can be described in Eq. (5).

$$\frac{y_i(w \cdot x_i + b)}{|w|} \quad (5)$$

The displayed distance must be at least the value of ρ as shown in Eq. (6).

$$\frac{y_i(w \cdot x_i + b)}{\|w\|} \geq \rho, \quad \forall i \quad (6)$$

Fixing ρ as $\rho = \frac{1}{\|w\|}$, the constraint given by $\forall (x_i, y_i) \in T, y_i(w_T x_i + b) \geq 1$ but the maximized expression becomes $\frac{2}{\|w\|^2}$. This is equivalent to minimizing $\frac{1}{2} \|w\|^2$, which defines the optimization problem as shown in Eq. (7).

$$\arg \min_w \frac{1}{2} \|w\|^2 \quad (7)$$

So this corresponds to the standard SVM formulation given as a minimization problem: find w and b such as Eq.(8).

$$\frac{1}{2} \|w\|_2^2 \text{ is minimized and } \forall (x_i, y_i) \in T, y_i(w \cdot x_i + b) \geq 1 \quad (8)$$

The optimization problem obtained is quadratic subject to linear constraints. Because the objective function is convex and the points that satisfy the constraint form a convex set, this problem presents a unique local minimum (Haykin, 2008). This can be solved by introducing a Lagrangian function whose purpose is to construct a dual problem in which the Lagrange multiplier α_i is associated with the constraint $y_i(w_T x_i + b) \geq 1$

in the primal problem to include them in the objective function (Manning; Raghavan; Schütze, 2008).

The dual problem corresponds to finding $\alpha_i, \dots, \alpha_N$ such as Eq. (9).

$$L(\alpha, w, b) = \frac{1}{2} ||w||^2 - \sum_{i=1}^n \alpha_i (y_i(w \cdot x_i + b) - 1) = \frac{1}{2} ||w||^2 - \sum_{i=1}^n \alpha_i (y_i(w \cdot x_i + b)) + \sum_{i=1}^n \alpha_i \quad (9)$$

The Lagrangian function must be minimized, which implies minimizing w and b and maximizing the variables α_i . There is a saddle point at which the gradients of the above equation with respect to w and b are zero and that $\alpha_i \geq 0$ as shown in Eq. (10).

$$\frac{\partial L}{\partial w} = 0 \Rightarrow w = \sum_i \alpha_i y_i x_i \quad \frac{\partial L}{\partial b} = 0 \Rightarrow \sum_i \alpha_i y_i = 0 \quad (10)$$

Inserting these equations, we have Eq. (11).

$$C \arg \max_{\alpha} \sum_j \alpha - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \quad (11)$$

Within the constraints in Eq. (12).

$$\begin{cases} \sum_i \alpha_i y_i = 0 \\ \alpha_i \geq 0, \forall i = 1, \dots, n \end{cases} \quad (12)$$

Once α_i is found, it is possible to determine w through the principal equation. There are three important properties: First, the expression is convex. Second, the data enters the expression only in the form of inner products of example pairs. Finally, with α determined, it is noted that most α_i are zero, but those with nonzero values indicate that the corresponding x_i is a support vector.

The second property also holds for the discriminant equation corresponding to Eq. (13).

$$h(x) = \text{sign}(\sum_l a_l y_l (x_l \cdot x) + b) \quad (13)$$

The support vectors are the examples x_i that satisfy $y_i(w \cdot x_i + b) = 1$ and are located on the edge. This fact can be used to calculate b from any support vector using $b = y_i - w \cdot x_i$. For numerical stability, this should be done for all support vectors, taking the average value of b . The resulting discriminant function is named a Support Vector Machine.

One of the major challenges, in statistical learning machines, concerns finding a kernel that optimizes the decision boundary between the classes of a given application. In ELM neural networks, a Linear kernel, for instance, is capable of solving a linearly separable problem, as seen in 5 (a). Following the same reasoning, Sigmoid, RBF, and Sinusoidal kernels are capable of solving problems separable by Sigmoidal, Radial, and Sinusoidal functions, seen in Fig. 5 (b), in Fig. 5 (c), and in Fig. 5 (d), respectively.

This way, the generalization capacity of a good neural network may depend on a well-adjusted choice of the kernel. The best kernel may be subordinate to the problem being solved. As a side effect, investigating different kernels is generally a costly process involving cross-validation combined with different random initial conditions. However, the investigation of distinct kernels may be necessary; otherwise, the composite neural network, by an ill-adjusted kernel, may generate unsatisfactory results. As a counterexample, observe the employment of the Linear kernel applied to Sigmoid and Sinusoidal distributions presented in Fig. 6 (a) and in Fig. 6 (b), respectively. The classification accuracies exposed in Fig. 6 (a) and in Fig. 6 (b) are 78.71% and 73.00%, respectively. Visually, it is possible to observe that the Linear kernel does not map the decision boundaries of the Sigmoid and Sinusoidal distributions adequately.

A good generalization capability of these kernels also depends on a well-adjusted choice of parameters (C, γ) . The cost parameter C refers to a reasonable balance point between the width of the hyperplane margin and the minimization of the classification error concerning the training set. The kernel parameter γ controls the decision boundary as a function of the classes (Lima, 2021, 2022a, 2022b 2023). There is no universal method in the sense of choosing the parameters (C, γ) . In the present work, the parameters C and γ vary exponentially in increasing sequences, mathematically according to the function 2^n , where $n = \{0, 5, 10\}$. The hypothesis is to verify if these distinct parameters from the standards; $(C, \gamma) = (2^0, 2^0)$ are capable of generating better accuracies.

Figure 5: Successful performances of *kernels* compatible with the datasets, these are authorial hypotheticalal educative examples

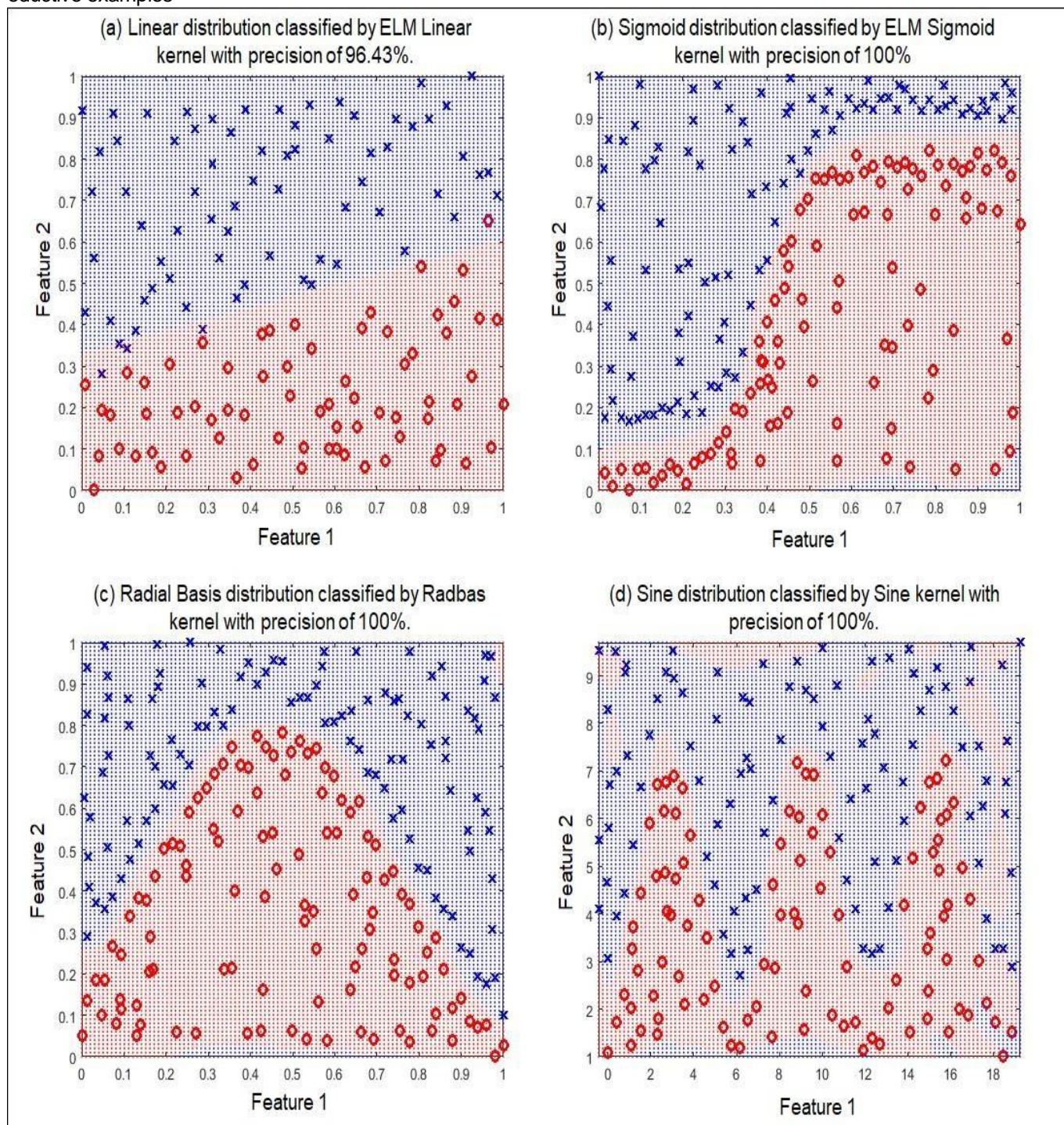
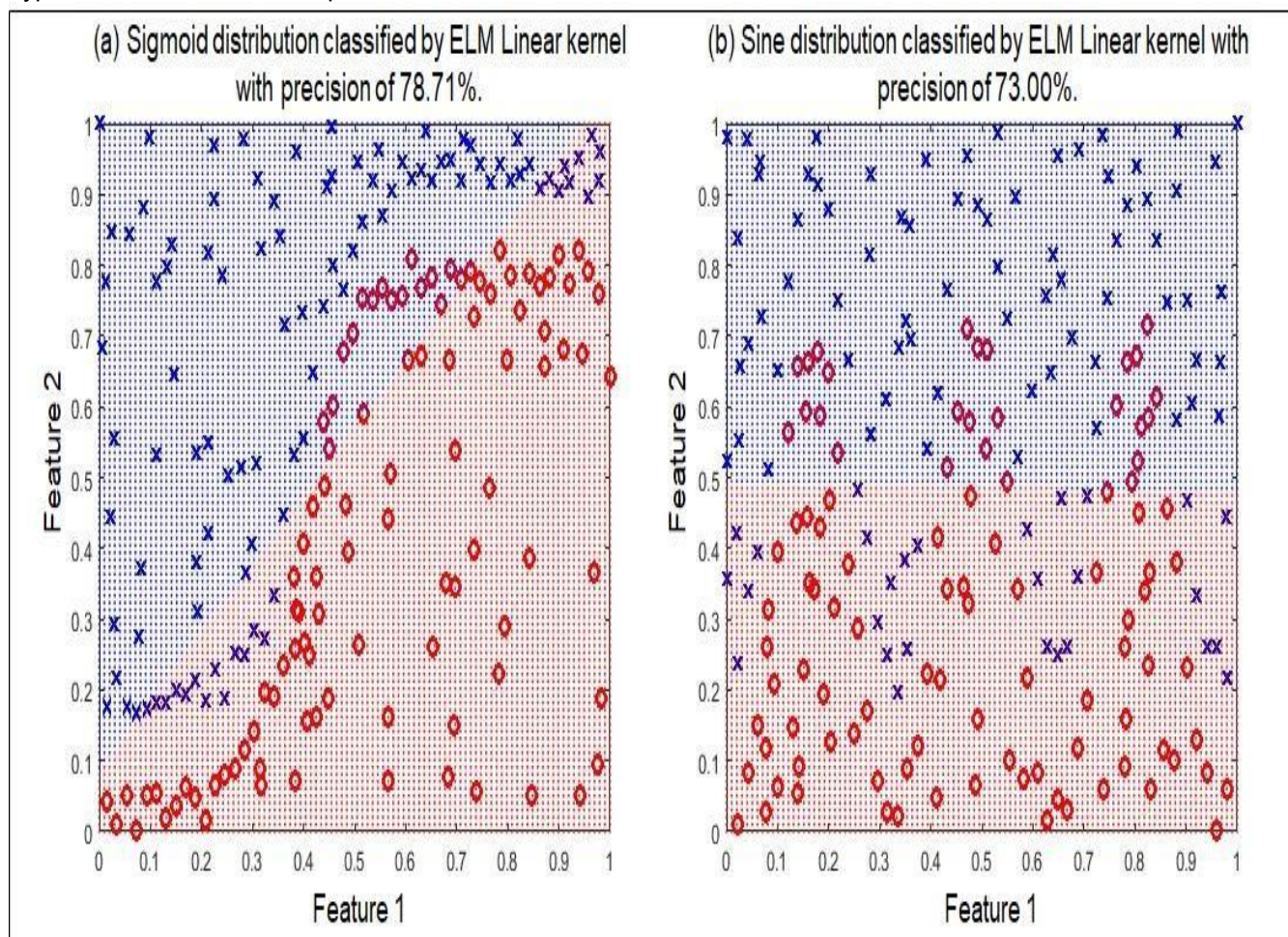


Figure 6: Unsuccessful performances of the Linear *kernel* on non-linearly separable datasets. These are authorial hypothetical deductive examples



MATERIALS AND METHODS

The present work uses two scenarios aimed at pattern recognition of clusters belonging to the target class. In the first scenario, here named “simple”, the classification is binary. There is only class (JPEG) vs. counter-class (PNG). This methodology was developed by Pavel (2017) and replicated in the aforementioned simple scenario.

In a second scenario, named “complex”, the one-against- all method was used. According to Table 3, the target class (JPEG) is dispersed in a hodgepodge of counter-classes. The creation of the complex scenario aims to establish a simulation of the machine of a certain common user. This user has a tendency to have a large amount of files for the most diverse purposes. The purpose of the work is to bring the experiment closer to the common use of personal computers by contemporary society.

All files are freely available in the copyright repository (Dejavu, 2024). The objective is that the proposed work can be replicated by third parties. It also aims to ensure the

veracity of the results achieved by the proposed methodology.

SIMPLE SCENARIO: BINARY CLASSIFICATION

In the presented scenario, 1,000 (thousand) JPEG files and 1,000 (thousand) PNG files were designated, respectively considered class and counter-class. The files were cataloged on the world wide web.

The proposed work creates an authorial tool aiming at target class pattern recognition for JPEG files. From a training set, it is possible to formulate a hypothesis about the clusters of the target class (JPEG). It is up to the authorial system to estimate the class of an unprecedented cluster. It does this by comparing its features audited in real time and those obtained during the training stage, The intention is that the proposed work can be replicated by other researchers, strengthening the verifiability of the results achieved by the proposed methodology. All experiments were carried out on a computer equipped with an 11th generation Intel(R) Core(TM) i5-11400H processor, with a speed ranging from 2.70GHz to 2.69GHz. There is 40.0 GB of RAM, 2 TB. There's a GeForce GTX 1650 graphics card.

COMPLEX SCENARIO: ONE-AGAINST-ALL METHOD CLASSIFICATION

In this scenario, a database was formed containing 16 thousand files, one thousand of each extension presented below. The files were cataloged on the world wide web. The main objective of the experiment is to simulate a common user's computer. In contemporary times, electronic computers such as desktop computers, cell phones and other devices are not just focused on entertainment. They became indispensable for the fulfillment of domestic, professional, academic and planning tasks.

The hypothesis is that the authorial system is able to differentiate clusters of the target class (JPEG) from other types of files, even though they were not presented during its training phase. The one-against-all method is applied in this complex scenario. In statistical terms, it only matters the segregation of clusters of the target class to the detriment of all other classes.

Table 1 - Complex Scenario: Authorial Dataset.

Class	Counter-class	Description
JPEG	PNG	Image formats commonly used for digital pictures.

Word (doc/docx)	Writer - LibreOffice (ODT)	Text editors for creating documents, papers, etc.
Excel (xls/xlsx)	Calc - LibreOffice (ODS)	Spreadsheet programs for data management and calculations.
PowerPoint (ppt/pptx)	Impress - LibreOffice (ODP)	Applications for creating presentations for seminars, projects, etc.
Access (accdb)	SQL	Database management systems.
Outlook (msg)	Gmail (EML)	Email file formats used for communication.
OneNote (one)	Joplin (JEX)	Virtual notebooks for organizing notes and tasks.
Publisher (pub)	TIFF	Tools for designing flyers, cards, and other visual media.

METHODOLOGY

EXPERIMENTS AIMED AT RECOVERING FORMATTED DATA

All files were allocated on a flash memory device - pendrive, and a prior logical sterilization was performed on this device. The fdisk command lists the partitions recognized by the operating system.

If the partition is not listed, digital forensic analysis will be unfeasible in our materials and methods. This is due to the Linux distributions' tools for digital forensic analysis. For example, the native data collection tool - dd: native data collection tool as Kali and Parrot Linux - in Linux distributions only works if the operating system lists the partition.

The main problems related to the partition not being listed correspond to the driver not being recognized. Another possibility is that the device is partially or completely broken. Next, logical sterilization reorganizes the device in order to define the file system, i.e. it works like cleaning up the directories on the device. This procedure was carried out using the wipe command:

- To locate the device directory:

fdisk -l

- Sterilization process:

wipe/dev/sda

After logical sterilization, all class vs counter-class samples are copied to the memory device. Subsequently, a simple formatting is performed, which does not consider logical sterilization. This procedure can be performed through the exclusion itself by graphic interface.

In the sequence, there is data collection. This collection corresponds to forensic duplication, that is, a bit-by-bit copy of the content found on the device being investigated. The file with the extension dd. which refers to the image of the target device.

Before court, forensics must be performed in this extension that works as a copy of the data. The electronic device must be returned to the defendant as soon as possible. The defendant must have ample right of defense. For this, it is necessary to return your electronic equipment after copying the image of the target device.

Furthermore, the constant use of the storage device can reduce its useful life partially or completely. This depletion is known as bad blocks. The stress promoted by a forensic investigation is emphasized by promoting a bit-by-bit scan of the electronic artifact. In line with the reduction in the useful life of the device, the possible material evidence of an investigation may be lost or compromised.

Exceptions may occur, depending on the magistrate's and tech staff's interpretations. In some cases, authorities must forfeit or destroy a device. This is partly because returning it would be an offense due to its stored material.

For data collection, the Brazilian Symposium on Information Security recommends the dd: disk dump software.

The Brazilian Information Security Symposium recommends the use of dd (disk dump) software for data collection. By adopting dd, our aim is for our tool to be used by police forces and accepted by the Brazilian judicial system. Unfortunately, Brazil and many South American countries, such as Argentina, are part of a route for malicious data, such as botnets (Research, 2023). Many malicious activities are stored on Brazilian servers. We intend to make Brazilian forensic practice recognize our tool by adopting dd. To this end, the support of the Brazilian Information Security Symposium is essential.

The dd parameters correspond to:

- -if: investigated partition;
- -of: collected image (bit-by-bit copy) of the investigated partition.
- In the terminal, we use:

```
dd if=/dev/sda1 of=image.dd
```

It should be noted that dd does not carry out verbose, that is, it does not print any information on the terminal. For the user, apparently, it results in the feeling that there was a failure in the processing. After the collection, the following formatted data recovery tools are employed.

Foremost

Foremost is a terminal software whose objective is to recover data formatted through Data Carving, considering the headers, footers and file structures, being able to work with image files or directly in a given unit. We use Foremost version 1.5.7, released on 07/12/2012.

- -t: specifies the type of files to be recovered, for example, JPEG, gif, png, bmp, or you can use all to recover all types of files.
- -i: abbreviation for input, for the source partition.
- -o: short for output, for the destination path of the files to be retrieved (on another partition such as an auxiliary memory device). By default, foremost creates a folder named output, if the user does not define the destination.
- In the terminal, we use Foremost:
`foremost -t all -i image.dd`

Scalpel

By default, all types of files are contained in the standard configuration file (systems/etc/scalpel/scalpel.conf files). This file contains comments that correspond to the configuration pattern. To specify which are the types of files to be extracted, it is necessary to uncomment the lines referring to the extensions of the files. We use Scalpel version 1.60, released on 6/27/2013.

- The source partition. Note that there is no -i directive even though the Scalpel documentation states that it is required.
- -o: short for output, for the destination path of the files to be retrieved. Its folder must first be created before Scalpel is invoked.
- -c: configuration file discussed earlier. At the terminal, we use Scalpel:
`scalpel image.dd -o /output2/
-c /etc/scalpel/scalpel.conf`

Magic Rescue

Magic Rescue also employs Data Carving to recover formatted data. By default, all types of files to be recovered are contained in: (files systems/usr/share/magicrescue/recipes) We use Magic Rescue version 1.1.10, released on 11/24/2018.

- -d: to the destination path of the files to be recovered. It is necessary that the folder is created before Magic Rescue is invoked.
- There is no flag (directive) for the source partition, just quote it in the command.
- -r: configuration files discussed earlier. In the terminal, use Magic Rescue:
magicrescue -d output3 image.dd -r avi
-r canon-cr2 -r elf -r flac -r gpl -r gzip
-r jpeg-exif -r jpeg-jfif -r mbox
-r mbox-mozilla-inbox -r mbox-mozilla-sent
-r mp3-id3v1 -r mp3-id3v2 -r msoffice
-r nikon-raw -r perl -r png -r ppm
-r sqlite -r zip

Photorec

PhotoRec is an open-source application. It has the function of recovering data that cannot be opened, and can be used on mobile devices such as pendrive, CDs and HDs. We use PhotoRec version 7.2, released on 02/22/2024.

Recuva

Recuva allows you to recover files that have been deleted on Windows system. This recovery is not restricted to the hard disk, it also makes it possible to rescue files saved on portable devices. Its main function is to locate files that can be recovered. However, the program also allows a complete deletion of files. We use Recuva version 1.53.2096, released on 06/13/2023.

Autopsy

Autopsy makes it possible to recover deleted data. Autopsy makes it possible to recover deleted data. With this tool, you cannot directly access the drive or image to perform Data Carving. You must follow the process of creating the case, building the preview file, the result files, analyzing the results, performing integrity checks, and extracting the data. We use Autopsy version 4.21.0, released on 09/06/2023.

Diskgenius

DiskGenius is used to recover deleted, lost or formatted files from a range of storage

devices, including hard drives, external HDDs, USB flash drives, virtual disks, memory cards and RAID arrays. We use DiskGenius version 5.6.1.1580, released on 08/15/2024.

Deca

The DECA employs machine learning for pattern recognition of clusters in JPEG files (Gladyshev; James, 2017). In the feature extraction phase, DECA constructs a histogram of the evaluated cluster. Then, in the classification phase, the cluster histogram serves as input neurons for the statistical learning machine. DECA does not employ distinct pattern recognitions to identify headers and footers. The machine learning is unique.

Regarding pattern recognition, DECA assumes a linear kernel. The mentioned kernel functions effectively when distributions are linearly separable. Hence, as an analysis method, it does not delve deeper into different learning functions. Different cost parameters and kernel parameter variations are not explored.

Parameters of the DECA tool:

- -vv: shorthand for verbose, which prints the progress of the examination on the screen.
- -o: shorthand for output, for the destination path of files to be recovered (on another partition such as an auxiliary memory device). It's necessary for the folder to be created before DECA is invoked.
- There's no marking (directive) for the source partition, just mention it in the command.
- -linear: without machine learning, just Data Carving.
- -deca: with machine learning added to Data Carving.
- -m jpeg.model: file containing the configuration parameters of the machine learning.
- On the terminal, we use DECA as follows:
`./deca -vv -o /home/kali/Desktop/output`
`--deca -m jpeg.model image.dd`

Dejavu Forensics: Technique Proposed by The Autors

Dejavu Forensics uses the following libraries:

- libtsk-dev (sleuthkit): responsible for reading clusters from the target storage device.

- libmagic-dev: Data Carving Manager (magic numbers).
- liblinear-dev: responsible for the pattern recognition step using linear discriminant.
- libsvm-dev: responsible for the pattern recognition stage using the Support Vector Machine.

Both Autopsy and Dejavu Forensics use Sleuth Kit to process a partition's clusters. But Dejavu's big differentiator is its integration with machine learning. Sleuth Kit is vital for data recovery in both tools. But Dejavu goes further. It uses machine learning to recover data and find patterns in the clusters. This approach allows for more accurate and efficient retrieval. It differs from traditional techniques by using machine learning on the retrieval process. This makes it compatible with pattern-based analysis.

The suggested technique uses machine learning to spot cluster patterns in JPEG files, specifically using the Support Vector Machine (SVM). In the feature extraction phase, the tool creates a histogram of the analyzed cluster. Later, during the classification step, this histogram acts as an input feature set for the statistical learning model.

The foundation of this tool is based on the DECA methodology, a Data Carving algorithm that combines the identification of specific digital signatures, known as "magic numbers", and cluster pattern recognition through machine learning. DECA was originally designed to efficiently recover non-fragmented JPEG data. This involves identifying clusters with JPEG data, done by detecting their header and footer digital signatures.

Moreover, the Dejavu tool uses a machine learning strategy to recognize JPEG file patterns. In this context, the Support Vector Machine (SVM) is employed, which has proven effective in recognizing patterns in complex datasets. SVM classification works on the principle of creating an ideal separation between classes. The main goal of the SVM is to stratify data by creating an ideal decision surface, called a hyperplane. This decision surface aims to optimize the accuracy of training set classification, ensuring the best margins relative to the support vectors. The underlying idea is that the optimal hyperplane will have a better generalization capability when applied to the test set.

Dejavu Forensics uses the following parameters:

- -vv: This option stands for verbose. It displays the progress.
- -o: This option means output, showing where to save the recovered files. The destination folder should be made before running the command.
- -dejavu: This option adds machine learning to the Data Carving process.

- `-oneclass`: This means the analysis will be done in a single mode, or for one file class.
- `-fex "raw"`: This says the feature extraction method used is raw, referring to direct data extraction from files. `histo` could be chosen when input features are about the cluster's histogram.
- `/dev/sdb1`: This is the device or partition to be checked, in this case, `/img.dd`.
- In the terminal, we use Dejavu as follows:

```
./dejavu -vv -o /home/kali/Desktop/Dejavu/output
-oneclass -fex "raw" img.dd
```

These commands show different settings and approaches in the Dejavu tool for PNG and JPG file recovery. Additionally, the `no footer` option controls whether machine learning is used to recognize footer patterns in the analysis results. We filed the first patent for Dejavu in 07/26/2023.

Dejavu differs from the existing DECA tool in several ways. Dejavu can identify both PNG and JPG files, whereas DECA can only detect JPG files. In pattern recognition, DECA uses only the linear kernel. As Figure 6 shows, linear kernel performs poorly on non-linearly separable distributions. On the other hand, Dejavu exploits a variety of kernels, including linear, polynomial, sigmoidal and radial. For each kernel, we investigate different initial conditions.

These include kernel parameters and gamma (related to curvature). Our framework also explores different feature extraction methods. These include the raw cluster and the cluster histogram. DECA can only analyze the cluster histogram. Although Dejavu Forensics was developed based on the methodology proposed by the DECA tool (2017), we are not limited to it. We expanded the approach to recovering formatted data using machine learning, which allowed us to achieve superior results in terms of accuracy and recovery time. We chose to compare Dejavu Forensics with popular commercial tools, like Recuva, Autopsy, and DiskGenius. This shows its relevance today. We aimed to do more than just test academic tools. In addition, section 6. The paper includes a comparison with commercial data recovery tools. It details the results and highlights Dejavu Forensics' superior performance.

RESULTS

Given the importance of recovering files from formatted devices, this study aims to

examine the performance and limitations of Data Carving, used to recover data of formatted devices, which involves identifying the initial signatures (headers) and endings (footers) associated with the corresponding file extensions.

The underlying premise is that each file type has a characteristic structure of bytes at the beginning and end. However, the approach based on the identification of headers and footers, called the "magic number", can lead to challenges, such as: (i) the generation of a large number of false positives, i.e. non-existent files; (ii) the accidental deletion of existing files; and (iii) the recovery of corrupted files containing only fragments of data.

The use of machine learning emerges as a promising alternative for overcome the limitations of Data Carving. Although machine learning is widely known and applied in several computational fields, its use in the digital forensic area is still is at an early stage.

Tables 2, 3, 4 and 5 present the results obtained by different analysis tools. recovery of formatted data.

Tables 4 and 5 show the results for recovering files in PNG format. Note that DECA was not included in these results. DECA can only detect JPEG files, as Tables 2 and 3 describe.

The tools used were Foremost, Scalpel, Magic Rescue, Photorec, Recuva, DiskGenius, Autopsy, DECA and Dejavu Forensics, developed in the present study. Although state- of-the-art tools are functional, they do not employ machine learning. On the other hand, Dejavu Forensics employs machine learning as an approach to overcome the limitations of Data Carving.

When analyzing the tables the "Magic N" terms refer to "magic numbers". These are sequences of bytes that indicate the format of the files. While the term "ML"(Machine Learning or Learning of Machine) is used to identify tools that use this approach for data recovery.

Table 2 presents the results obtained by each software in the Simple Scenario. Restricted to formatted JPEG and PNG files. Scalpel had a worrying problem, with a 99.90% false positive file rate, meaning that among the files it recovered, only 1 was actually true, and the others were recovered wrong. Magic Rescue also had a high percentage of false positive files, reaching 96.08%, and still generated repeated files.

As for the generation of repeated files during the process, Autopsy stood out negatively in this aspect. Although it achieved with 100% accuracy in relation to the original data, Autopsy generated almost double the number of repeated files, which resulted in 999

more files than the original data. Therefore, Autopsy generated 49.97% of duplicate files, although true and belonging to the database of data. By establishing a relationship between execution time and recovery of true positive, Dejavu Forensics recovers 100% of formatted files in just 7 seconds.

DiskGenius, in the simplest scenarios, the tool proved to be effective, especially for JPEG files, where it achieved an accuracy of 80.10%. However, in the simplest scenario for PNG files, the tool performed less impressively, with an accuracy of 40.70% against the database and 76.79% for true files. This was accompanied by a false positive rate of 7.55% and a duplication rate of 15.66%.

Table 3 and 5 presents the results obtained in the Complex Scenario. There are 16,000 files with 16 extensions widely used by common users. When we look at the possibility of false positive results, we see that the Dejavu and Recuva did not generate any false positive files. In isolation, the Dejavu tool stands out, which, in addition to ensuring a full recovery rate of 100% of files, performed the operation in a remarkable time of 13 seconds. Dejavu Forensics was unaware of any manifestations of false positive and/or repeated files.

Table 4 refers to data recovery in PNG format in the simple scenario. Analysis of the tools revealed that most tools achieved accuracy levels above 96%, excluding Scalpel, which was unable to recover any files. Regarding the generation of repeated files, only Magic Rescue and Autopsy generated duplicates, recording proportions of 42.86% and 49.85%, respectively. This finding denotes that, despite of acceptable accuracy, the two software programs generated almost half of the data files repeated form.

Regarding processing time, once again, Dejavu stands out, being able to recover formatted data in just 9 seconds. It is worth noting that, in the scenario under examination, Dejavu Forensics demonstrated an accuracy of 98.27%, with the incidence 1.73% of erroneously generated files, proving to be an effective and agile alternative.

Table 5, the results achieved in the complex scenario are shown, comprising an amalgamation of 16 thousand files that aims to emulate the behavior of a user common. In terms of generating repeated files, Foremost generated 17.75% of files repeated, while Magic Rescue had a repeat rate of 20.55%. One more Once again, Autopsy draws attention because, despite boasting an accuracy close to 99.90 in the recovery of true files, it presented a rate of 60.65% of repeated files. Even in a scenario that includes 16 thousand files, Dejavu managed to rescue 98.10% PNG extension files.

General, the Recuva and Photorec tools draw attention for their accuracy high and low incidence of false positive and repeated results. Furthermore, the tool Dejavu Forensics stands out for its accuracy in file recovery and, mainly, for its ability to be efficient in an extremely short time frame compared to other tools. Dejavu Forensics establishes the employment of machine learning and science data as a promising path in the field of digital forensics.

Table 2 - JPEG Simple scenario: tool results.

Tool	Comp. Type	Num. of gen. files	Runtime	Size of dir. (MB)	False pos. (%)	True duplicates (%)	True pos. from db (%)	True pos. (%)
Dejavu Forensics	Magic numbers + Machine Learning	1000	7 s	39.9 MB	0	0%	100%	100%
Deca	Magic numbers + Machine Learning	1005	18,96 s	40.1 MB	0.10%	1.39%	99%	98.51%
Recuva	Magic numbers	996	5m 12s	39.8 MB	9%	0%	90.60%	91%
Foremost	Magic numbers	998	10m 53s	39.8 MB	0,20%	0%	99.60%	99.80%
Scalpel	Magic numbers	1000	8m 17s	4600 MB	99.90%	0%	0.10%	0.10%
Magic Rescue	Magic numbers	1555	7m 22s	202.5 MB	96,08%	1.16%	4.30%	2.77%
Photorec	Magic numbers	999	4m 39s	39.9 MB	0%	0%	99.90%	100%
Autopsy	Magic numbers	1999	34m 17s	584.6 MB	0%	49.97%	100%	50.03%
DiskGenius	Magic numbers	831	2m 45s	258 MB	3,13%	0,48%	80.10%	96,39%

Table 3 - JPEG Complex scenario: tool results

Tool	Comp. Type	Num. of gen. files	Runtime	Size of dir. (MB)	False pos. (%)	True duplicates (%)	True pos. from db (%)	True pos. (%)
Dejavu Forensics	Magic numbers + Machine Learning	1000	13 s	39.9 mb	0%	0%	100%	100%
Deca	Magic numbers + Machine Learning	1002	33,21 s	40 mb	0.20%	0%	100%	99.80%
Recuva	Magic numbers	1000	1h 14 m 21s	39.9 mb	0%	0%	100%	100%
Foremost	Magic numbers	4902	11m 39s	403.1 mb	61.87%	17.75%	99.90%	20.38%
Scalpel	Magic numbers	412	13m 57s	1400 mb	100%	0%	0%	0%

Magic Rescue	Magic numbers	798	2h 43m	51.4 mb	78.32%	20.55%	0.90%	1.13%
Photorec	Magic numbers	1000	6 m 34s	40.1 mb	1.80%	0%	98.20%	98.20%
Autopsy	Magic numbers	2695	2h 27min 12s	108.5 MB	0.04%	62.86%	100%	37.11%
DiskGenius	Magic numbers	830	3min 28s	24.8 MB	0,12%	0%	82.90%	99.88%

Table 4 - PNG Simple scenario: tool results.

Tool	Comp. Type	Num. of gen. files	Runtime	Size of dir. (MB)	False pos. (%)	True duplicates (%)	True pos. from db (%)	True pos. (%)
Dejavu Forensics	Magic numbers + Machine Learning	985	9 s	252.4 MB	1.73%	0%	96.8%	98.27%
Recuva	Magic numbers	1000	5m 12s	252.4 MB	0.10%	0%	99.90%	99.90%
Foremost	Magic numbers	973	10m 53s	210.5 MB	0.31%	0%	97.00%	99.69%
Scalpel	Magic numbers	0	8m 17s	0	0%	0%	0%	0%
Magic Rescue	Magic numbers	1829	7m 22s	456.6 MB	2.79%	42.86%	99.40%	54.35%
Photorec	Magic numbers	998	4m 39s	252.4 MB	0.30%	0%	99.50%	99.70%
Autopsy	Magic numbers	1998	34m 17s	504.8 MB	0.15%	49.85%	99.90%	50.00%
DiskGenius	Magic numbers	530	1m 9s	8.68 MB	7.55%	15.66%	40.70%	76.79%

Table 5 - PNG Complex scenario: tool results.

Tool	Comp. Type	Num. of gen. files	Runtime	Size of dir. (MB)	False pos. (%)	True duplicates (%)	True pos. from db (%)	True pos. (%)
Dejavu Forensics	Magic numbers + Machine Learning	1010	13 s	260.4 MB	2.77%	0.10%	98.10%	97.13%
Recuva	Magic numbers	1001	1h 14m 21s	252.5 MB	0.20%	0%	99.90%	99.80%
Foremost	Magic numbers	4410	11m 39s	365.2 MB	38.96%	39.84%	93.50%	21.20%
Scalpel	Magic numbers	539	13 m 57s	4500 MB	82.75%	17.25%	0%	0%
Magic Rescue	Magic numbers	8060	2h 43m	83.3 MB	8.10%	90.58%	10.60%	1.32%
Photorec	Magic numbers	1000	6m 34s	252.5 MB	0.50%	0%	99.50%	99.50%
Autopsy	Magic numbers	2554	2h 27m 12s	632.1 MB	0.26%	60.65%	99.90%	39.12%
DiskGenius	Magic numbers	527	1m 58s	8.47 MB	20.87%	3.04%	40.10%	76.09%

COMPARISON WITH COMMERCIAL DATA RECOVERY TOOLS

Recovering formatted data is one of the main demands in the field of digital forensics. While popular tools such as Recuva, Autopsy and DiskGenius offer paid solutions, with publicly traded companies such as Gen Digital Inc. (owner of Recuva) which has over 3,400 employees. Dejavu Forensics has emerged as a free and open source alternative. Even when competing with large, robust companies like Gen Digital, Dejavu Forensics offers superior performance through the use of machine learning (Dejavu, 2024).

The main commercial developers of these tools are improving them. Their updates can be biannual, annual, or even more frequent. Section 4 discusses this update cycle in detail. It covers its impacts and uses in digital forensics. It highlights how these improvements affect the tools' performance.

Unlike traditional methods based only on File Carving, Dejavu Forensics uses support vector machines (SVM) to find block patterns and data clusters. Data Carving often causes corrupted files or false recoveries. This significantly reduces the rate of false positives. In tests, the tool recovered over 96% of PNG and JPEG files. It beat paid tools in accuracy and speed, completing recoveries in under 13 seconds. Another crucial point is the transparency and accessibility of the tool. The Dejavu Forensics source code is available to the community, allowing it to be replicated and adapted by other researchers and professionals in the field.

SVM CONFIGURATION PARAMETERS FOR OPTIMIZATION IN IMAGE RETRIEVAL

In image retrieval, people use Support Vector Machines (SVM). They have been proven to be effective (Gladyshev; James, 2017). SVM is a supervised machine learning model that can be applied for both classification and regression tasks. In this work, the focus is on image classification. It aims to distinguish between predefined categories of visual content.

Our study explored the standard parameters of SVM. It also explored a wide range of parameters to optimize the proposed solution. The goal was to increase its accuracy. We conducted a detailed investigation. It focused on the best way to set SVM parameters. The investigation specifically looked at how to retrieve formatted files.

In terms of feature extraction, two methods were explored: the cluster histogram and the raw cluster itself. In the realm of kernels, we investigated four distinct functions. They

were: Linear, Polynomial, RBF (Radial Basis Function), and Sigmoid. The parameters (C, γ) have exponential variations.

They increase according to the function 2^n , where $n = \{0, 5, 10\}$. The hypothesis is to verify whether these parameters, different from the standards; $(C, \gamma) = 2^0, 2^0$, can generate better accuracies. The statistical learning repository studied the class versus anti-class. It also studied single class methods. It is emphasized that investigating the cost parameter C makes no sense when only a single class is contained in the repository. The cost parameter C refers to a reasonable balance between the margin width of the hyperplane while weighing the classes. When there is only one class, there is no weighing between different classes. For each scenario, we investigated 400 configurations. There were 400: 2 extractions * 4 kernels * 5 C parameters * 5 γ parameters * 2 types of repositories.

Dejavu tool aims to recover from JPG images. Both for header and footer, the best parameter is in the raw cluster's feature extraction. It is in the search space for optimized configurations for JPG image retrieval, both for header and footer. In the header-specialist model for JPG files, the RBF kernel is used. It's known for its ability to handle nonlinear features. It has a fixed γ parameter of $2^5 = 32$.

This γ value directly influences the kernel configuration. It determines the RBF radial's curvature in relation to the data. For the footer-specialist model of JPG files, the optimized kernel was Polynomial, degree 3, with a γ parameter fixed at $2^5 = 32$. The best way to create the statistical learning repository is for one class. This configuration is generally useful in scenarios with a large set of data from one class. The aim is to detect deviations or anomalies from the class pattern.

Dejavu tool also aims to recover from PNG images. It kept the best feature extraction method: the raw cluster. However, the γ value was changed to $2^{10} = 1024$. This was a big increase. It reflects the greater complexity and variation of PNG images compared to JPG. In the header-specialist model for PNG files, the RBF (Radial Basis Function) kernel was used. As for the footer-specialist model of PNG files, the chosen kernel was a degree 3 Polynomial. The Polynomial kernel can capture polynomial patterns in the data. This kernel is useful in cases where the features of the images and their categories has a non-linear relationship. But it can still be modeled with polynomials. Both the header and footer models have the same methodology in repository setup. It considers a single class.

We adjusted the parameters after a series of experiments and analyses. We sought the best setup for finding images effectively. The Dejavu tool, used to implement the SVM,

provided a robust means to test and apply these settings. The achieved results show that the parameter settings were decisive. They incremented how well image retrieval worked for both JPG and PNG formats.

CONCLUSIONS

In a tech-heavy, media-saturated society, this work aims to clarify digital crimes. It focuses on recovering intentionally formatted data. This demand arises as we face increasingly sophisticated cybercrimes. When intentionally formatting a disk or device, file removal occurs primarily in a logical sphere, making the previously occupied space available for new files. However, it is crucial to highlight that the original data still remains physically located on the device. These are structured in clusters, each with different headers that signal the beginning of certain files or extensions. Such headers and footers are of utmost importance in locating the desired data in the recovery process.

Forensic expertise, given this scenario, can significantly benefit from automated mechanisms, especially with the incorporation of innovative Machine Learning solutions. The "Dejavu Forensics" tool is a testament to this innovation, demonstrating effectiveness by recovering over 96% of for-matted files such as PNG and JPEG in a matter of seconds.

The scientific-methodological rigor used here suggests that Dejavu Forensics can offer substantial contributions to the advancement of digital forensics, providing greater efficiency, precision and reliability in investigations.

All files and data used in this work are in the repo (Dejavu, 2024). This supports our commitment to transparency and replicable research. Also, this study is relevant for more than just data recovery. It stresses the need for tech to fight today's digital crimes, like money laundering, tax evasion, and pedophilia.

It is possible to state that the objectives of this research were achieved. This work not only presents a significant technological innovation, but also highlights the ability of technology to serve human rights, promoting justice and improving the quality of life in the digital age. Ultimately, it serves as a beacon, lighting the way for future investigations and solutions at the intersection of technology, justice, and society.

FUTURE WORK

Our machine learning was trained specifically for the png and jpeg formats. These formats were chosen due to their prevalence in digital forensic investigations and their

widespread use in electronic devices. Focusing the study on these file types allowed for a better test of the proposed machine learning method. They are representative of typical data recovery scenarios. Also, JPEG and PNG files are popular in forensics. Their structure lets us test machine learning techniques with them.

We are aware that other file formats are important in forensic investigations, so we plan to expand the tool to other file types in future versions. At this stage, our main priority has been to establish the reliability and certainty of the model with the most common formats.

Although our study has shown promising results in recovering data formatted in jpeg and png formats, we recognize that there are some important limitations. The proposed method only works on jpeg and png files. This limits the Dejavu Forensics tool's use on other file types with different structures.

Due to good results in retrieving png and jpeg data, we will expand the Dejavu Forensics tool. It will cover other file types used in forensics. We intend to train the model with a database containing 16 file types, including, in addition to jpeg and png, formats such as doc, odt, xls, ods, ppt, odp, accdb, sql, msg, gmail, one, jex, pub and tiff. This expansion will allow the tool to support a wider range of data recovery scenarios, providing a faster and more robust solution for cases involving different file formats. Training the tool with these 16 file types could greatly improve forensic investigations. It would help police and other agencies recover and analyze data faster and more accurately. A better file search can speed up cybercrime investigations, like fraud and data breaches. It can help find crucial evidence quickly.

Also, by adding these new file types and expanding the tool, we expect better use of Dejavu Forensics in complex investigations. The tool should help in many cases. These include recovering business documents, image files, messages, presentations, and databases. This will aid in solving digital crimes.

Another future aim is to extend the approach to the recovery of formatted data, even if it is compressed or incomplete. This advance is key for cybersecurity and digital forensics. In these cases, the preservation of evidence is vital.

REFERENCES

1. ALHERBAWI, N., SHUKUR, Z., SULAIMAN, R. Systematic literature review on data carving in digital forensic. *Procedia Technology*, 11, 86–92, mar. 2013.
2. ALI, R.R., MOHAMAD, K.M., JAMEL, S., KHALID, S.K.A. A review of digital forensics methods for jpeg file carving. *J. Theor. Appl. Inf. Technol.*, 96, 5841–5856, out. 2018.
3. CARRIER, B. *File system forensic analysis*. Addison-Wesley Professional, 2005.
4. CASEY, E. *Digital Evidence and Computer Crime: Conducting Digital Investigations*. 3. ed., Elsevier, Maryland, EUA, 2011.
5. DEJAVU. *Dejavu forensics*. URL: <https://github.com/DejavuForensics/version-1.0>, acessado em mar. 2024.
6. DUDA, R.O., HART, P.E., et al. *Pattern Classification*. John Wiley & Sons, 2006.
7. GARFINKEL, S. Anti-forensics: Techniques, detection and countermeasures, in: *2nd International Conference on i-Warfare and Security*, pp. 77–84, 2007.
8. GARFINKEL, S.L., SHELAT, A. Remembrance of data passed: A study of disk sanitization practices. *IEEE Security & Privacy*, 1, 17–27, jan. 2003.
9. GLADYSHEV, P., JAMES, J.I. Decision-theoretic file carving. *Digital Investigation*, 22, 46–61, jun. 2017.
10. HANNAN, M. To revisit: What is forensic computing?, in: *Australian Computer, Network & Information Forensics Conference*, pp. 103–111, 2004.
11. HAYKIN, S. *Neural Networks and Learning Machines*. 3. ed., Pearson, 2008.
12. INC., G.D. *Perfil e dados da empresa - Gen Digital Inc*. URL: <https://stockanalysis.com/quote/prs/GEN/company/>, acessado em mar. 2024.
13. JAMES, D. *Forensically unrecoverable hard drive data destruction*. Infosec Writers, 2006.
14. KENT, K., CHEVALIER, S., GRANCE, T. *Guide to integrating forensic techniques into incident response*. Tech. Rep. 800-86, 2006.
15. LAURENSEN, T. Performance analysis of file carving tools, in: *Security and Privacy Protection in Information Processing Systems: 28th IFIP TC 11 International Conference, SEC 2013, Auckland, Nova Zelândia, 8-10 jul. 2013*. Proceedings 28, Springer, pp. 419–433.
16. LIMA, S., SILVA, S.H.M.T., P.R.e. Next generation antivirus for javascript malware detection based on dynamic features. *Knowledge and Information Systems*, 2023.

17. LIMA, S., SILVA, S., PINHEIRO, R., et al. Next-generation antivirus endowed with web-server sandbox applied to audit fileless attack. *Soft Computing*, 2022. doi: <https://doi.org/10.1007/s00500-022-07447-4>.
18. LIMA, S., SOUZA, D., PINHEIRO, R., SILVA, S., et al. Next generation antivirus endowed with bitwise morphological extreme learning machines. *Microprocessors and Microsystems*, 81, 103724, dez. 2021. URL: <https://www.sciencedirect.com/science/article/pii/S0141933120308693>, doi: <https://doi.org/10.1016/j.micpro.2020.103724>.
19. MANNING, C.D., RAGHAVAN, P., SCHÜTZE, H. *Introduction to Information Retrieval*. Cambridge University Press, 2008.
20. HOSSEINZADEH, M., RAHMANI, A.M., B.V.M.B.M.M.M.Z. *Improving security using svm-based anomaly detection: issues and challenges*. Springer-Verlag GmbH Germany, part of Springer Nature, 2020.
21. (OCC), D.C.O. Sextortion. URL: https://occ.org.br/tipo_de_fraude/sextortion/, acessado em out. 2020.
22. PAL, A., MEMON, N. The evolution of file carving. *IEEE Signal Processing Magazine*, 26, 59–71, mar. 2009.
23. PINHEIRO, R., LIMA, S., SOUZA, D., et al. Antivirus applied to jar malware detection based on runtime behaviors. *Scientific Reports - Nature*, 12, 1945, 2022. doi: <https://doi.org/10.1038/s41598-022-05921-5>.
24. RESEARCH, E. ESET takes part in global operation to disrupt the Grandoreiro banking trojan. URL: <https://www.welivesecurity.com/en/eset-research/eset-takes-part-global-operation-disrupt-grandoreiro-banking-trojan/>, acessado em 17 set. 2024.
25. SARI, S.A., MOHAMAD, K.M. A review of graph theoretic and weightage techniques in file carving, in: *Journal of Physics: Conference Series*, IOP Publishing, p. 052011, 2020.
26. SENCAR, H.T., MEMON, N. Identification and recovery of jpeg files with missing fragments. *Digital Investigation*, 6, S88–S98, jan. 2009.
27. RUSSELL, S., NORVIG, P. *Artificial Intelligence*. 2013.
28. VACCA, J.R. *Computer Forensics: Computer Crime Scene Investigation*. 2006.
29. WITTEN, I.H., FRANK, E., HALL, M.A., PAL, C.J., DATA, M. *Practical Machine Learning Tools and Techniques*, in: *Data Mining*, 2005.