Chapter

Information Hiding and Copyrights

Istteffanny Isloure Araujo

Abstract

This chapter explores the use of steganography on digital files and produces an enhanced technique that addresses the major vulnerabilities that make algorithms less reliable in securing data. Through a review of historical techniques in the field, the study identifies weaknesses in the algorithms to improve security and increase capacity using different techniques. One of the approaches proposed in this study involves a distributed method, which is simple, clear, low-cost, and agile. The study also analyses data manipulation and embedding processes in different files and for different purposes, such as vulnerabilities or placeholders exploited by criminals distributing viruses over the internet using Steganography. The results of the study can help forensic analysts identify secret content and raise awareness about protecting against eavesdropping data on devices. The study proposes a new scheme to improve Steganography called DSoBMP, together with guideline materials that have been published in four international peer-reviewed journals, including Springer and used as a stepping stone to collaborate in a worldwide book publication.

Keywords: forensics, information hiding, steganalysis, steganography, cryptography

1. Introduction

There is evidence that information hiding plays a pivotal role in regulating confidentiality in Cybersecurity. Steganography is a major area of interest within the field of Information hiding. Recently researchers have shown an increased interest in Copyrights. The main challenge faced by many researchers is the weaknesses of current algorithms to protect copyright data and issues such as low capacity to embed the information hiding or logarithm of copyrights to digital data files. Data from several studies suggested that capacity, detectability and distortion are the main issues in terms of using information hiding to protect copyright materials. A much-debated question is whether you can improve one area without compromising the other. Previous studies of information hiding have limited content and considerations with Big Data or even with studying different digital data files that can lead to a stronger technique. Up to now, too little attention has been given to improving all weaknesses of Steganography at the same time and in different files. Currently, there is no data on how to effectively stop applications such as snipping tools in all files, using different applications. This chapter deals with improvements of the main weaknesses of Steganographic Information hiding on different files using a combination of techniques and a distributive approach, we name this method Distributed Steganography over BMP phase I (DSoBMP-I) even though we can use it on different data files, the conversion to BMP type can increase capacity and improve many areas of Steganographic methods, such as flexibility to use it within other file formats. The specific objective of this research is to use intellectual property materials and apply copyrights. A qualitative and Quantitative research design was adopted, providing new insight into the distribution of the copyright content using a safer method that increases capacity, lowers detectability and minimizes distortion. The reader should bear in mind that still there are vulnerabilities within snipping tools and print screen techniques to be analyzed further. The experience of working with photographic content and social media led to the idea of applying a strong copyright that would follow the creative content without being easily broken by cyber criminals. The first session will examine file structures and information hiding applied to them, followed by recent work, copyright issues and the DSoBMP method and its implications.

2. Steganographic file structure exploits programmatically

Many information hiding methods are essential to either secure copyrights or identify malicious data embedded in files, like checking content on the end of the file tags on PDF documents - EOF or EOI at the end of the image files, plus checking on metadata for any other format - the description of the file, programmatically as well as the size of the file. False-positive events, meaning that there is no malicious content, simply a secure message or an error from the program are also valuable information to consider when trying to identify hidden content. The study of software that identifies malicious data is beneficial to understanding how to protect confidential information and copyrights, but it is difficult to get hold of the original code to reverse engineer it. It would be unsafe to disseminate it, as if cybercriminals get access, they could potentially uncover secret messages, see **Figure 1** for the File chunks, containing EOF and EOI to hide data.

We are not primarily focused on using steganalysis to identify crimes. Instead, we analyze and use the best carrier (image file format) along with Distributed Steganography to ensure the security of private messages and copyrights. Our proposed method of Distributed Steganography involves embedding confidential data over several different images generated from the original carrier of the stego-image. This technique provides a successful steganography tool to share images on the web without infringing on the author's copyrights [1]. We approach this method differently, as shown in **Figure 2**.

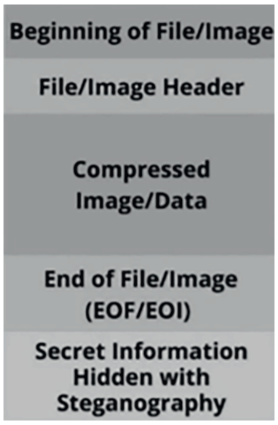

Beginning of File/Image
File/Image Header
Compressed Image/Data
End of File/Image (EOF/EOI)
Secret Information Hidden with Steganography

Figure 1.
EOF/EOI capabilities of hiding data.

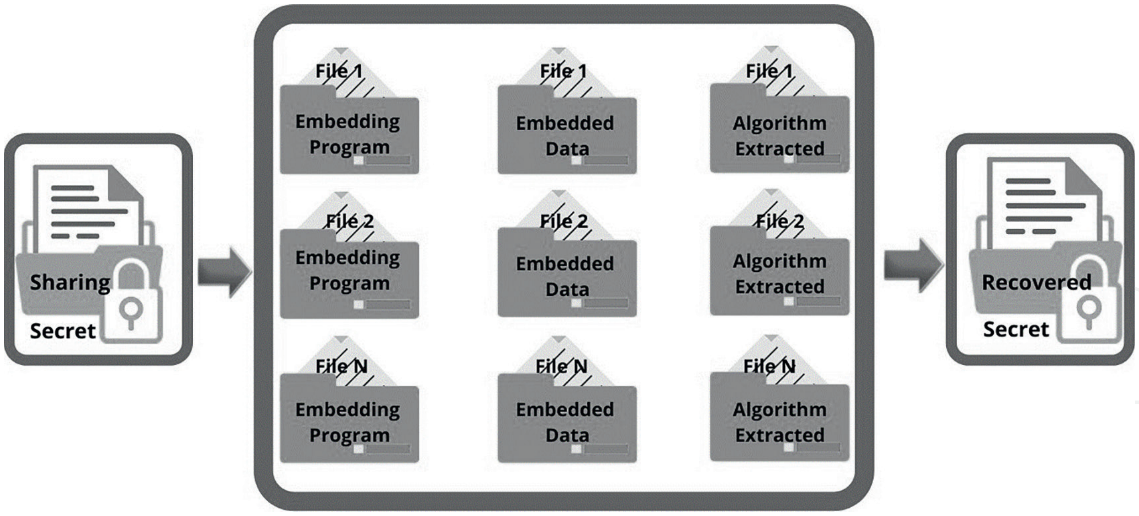
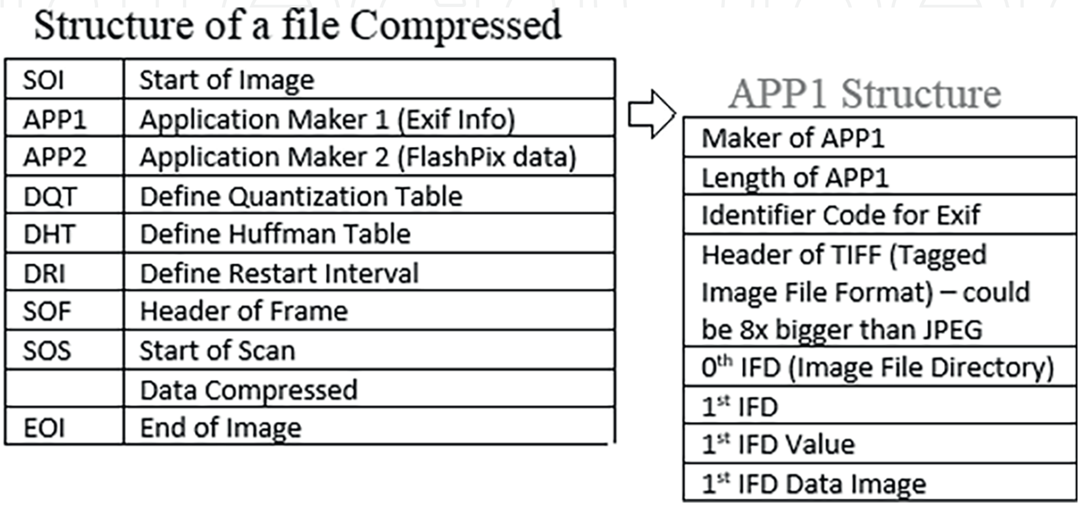


Figure 2.
Distributed steganography.

An image can be represented as a matrix of pixels, and the Spatial Domain of an image is simply the image itself. Image Steganography techniques that use the Spatial Domain method modify the pixel values of an image, but such techniques are not foolproof and can be vulnerable to steganalysis [2]. Private Key Steganography, on the other hand, involves using a key to embed and extract data [3]. It is possible to have the key generated automatically, eliminating the need for manual selection. However, dealing with different file formats and image structures can be complex. **Figure 3** demonstrates the stages involved in a JPEG-based technique, and each stage involves various algorithms and tasks, adding to the overall complexity [4].

The capacity of steganography algorithms refers to the amount of data that can be hidden within a carrier file, which in our case is an image while adhering to the limitations of the particular algorithm [5]. Capacity is also used to detect steganography in an image, as heavily modified bits can indicate the presence of hidden data. Therefore, capacity is a key metric for measuring the effectiveness of steganography algorithms.



Structure of a file Compressed

SOI	Start of Image
APP1	Application Maker 1 (Exif Info)
APP2	Application Maker 2 (FlashPix data)
DQT	Define Quantization Table
DHT	Define Huffman Table
DRI	Define Restart Interval
SOF	Header of Frame
SOS	Start of Scan
	Data Compressed
EOI	End of Image

APP1 Structure

Maker of APP1
Length of APP1
Identifier Code for Exif
Header of TIFF (Tagged Image File Format) – could be 8x bigger than JPEG
0th IFD (Image File Directory)
1st IFD
1st IFD Value
1st IFD Data Image

Figure 3.
Structure of JPEG.

Digital files with lower capacity tend to have higher detectability [6]. To understand the capacity of a file or image, we need to study the structure in detail. The structure of a JPEG file has its complexity and embedded compression techniques, they are not very proficient with Steganalysis. Algorithms with low capacity tend to introduce more distortion to steganographic files. That is why research on ways to improve capacity while minimizing detectability and distortion is highly valuable, given the significant impact these factors have on each other. Encryption can render a message unreadable by encrypting some or all of it, while Watermarking is used to add visible copyright messages, and Encryption is used to secure them invisibly.

Achieving higher capacity without compromising detectability and distortion is a challenging task indeed. It is important to conduct a steganalysis investigation to detect the presence of steganography. While there are several methods to detect steganography, observing distortion is the simplest way to do so. **Figure 4** shows different Steganalysis techniques such as visual, statistical, signature-based, spread spectrum and transform domain. Visual techniques are the most commonly used ones. If there is visible distortion, it becomes easier to determine if there is content hidden inside a file, however, visible distortion can also appear on images that are not formatted properly [7]. Therefore, combining more Steganalysis methods provides a better diagnosis. It is worth noting that some algorithms may have minimum distortion but still be detectable through statistical analysis.

The Least Significant Bits (LSB) algorithm is a technique used to hide data in a carrier file without affecting its quality. The hidden content is placed in the least significant bits of the file, which generally does not distort the file. However, the amount of data that can be hidden depends on the LSB capacity of the file. To detect the hidden data, a mathematical Steganalysis algorithm can be used. The LSB method is the most commonly used technique in the Spatial Domain category [8]. It can be used with any file format, but the detection process involves statistical analysis which begins by analyzing the spaces present in the file [9]. Mondal and Mandal's [10] experiments demonstrated how simple it is to hide information in the least significant bits. An example is shown in **Figure 5**. Histogram-based data hiding is a technique that involves inserting data into the highest frequency bits of an image. This method increases the image's robustness and can be reversible since it distributes personal data among the pixels with the highest frequency intensity. The process involves analyzing the intensity of the pixels within a black-and-white or color image, measuring their RGB values (red, green, and blue), as well as their brightness and contrast [11].

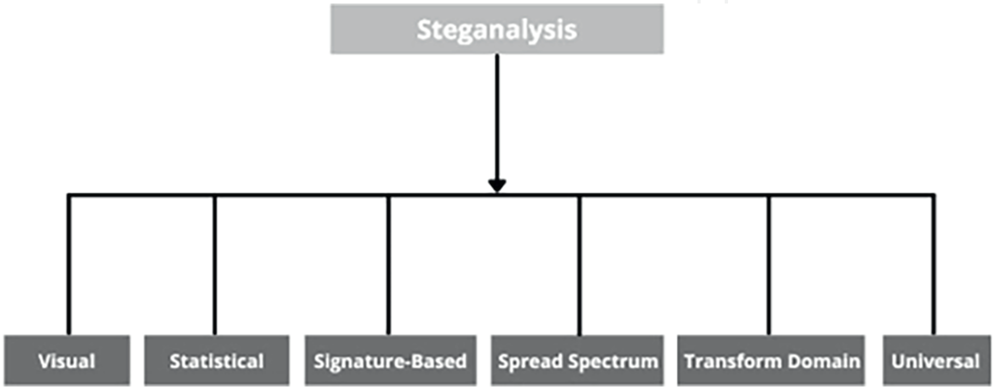


Figure 4.
Steganalysis techniques.

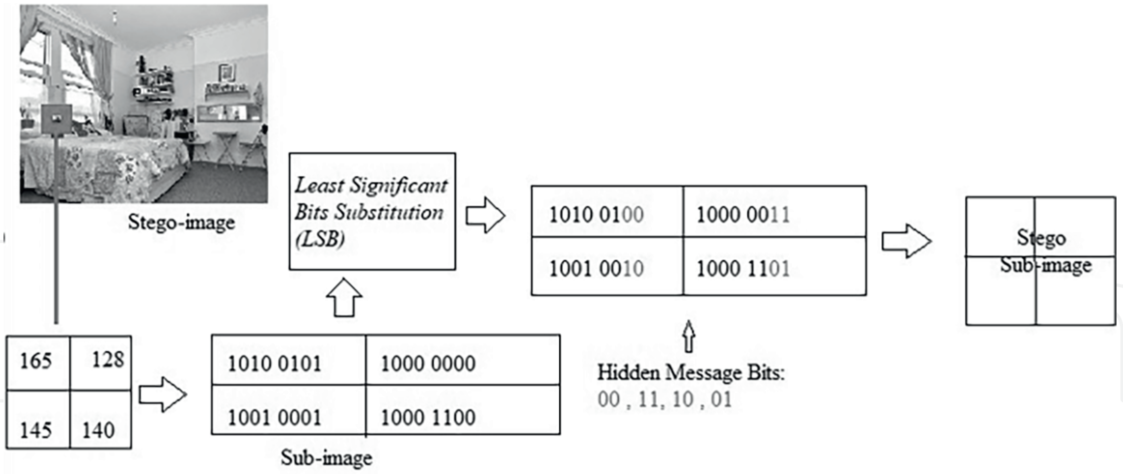


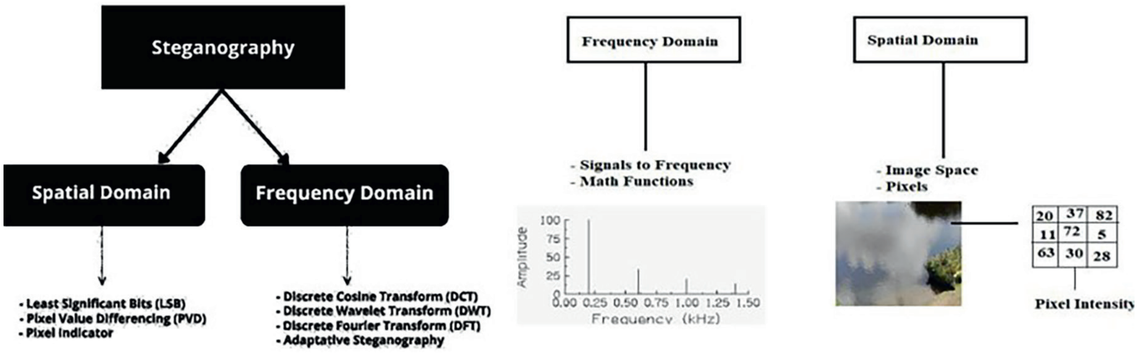
Figure 5.
LSB substitution on images.

By hiding data in specific colors and intensities of a stego image, the technique makes it possible to conceal personal information in a way that is difficult to detect.

The use of the Frequency Domain as a Steganographic technique involves processing the carrier image based on its transform, utilizing mathematical operations to refer to signals by frequency as depicted in **Figure 6**. This approach enhances data security by modifying the image through a variety of techniques outlined below, resulting in a distinct image containing embedded hidden data. Factors such as greyscales, content frequency, and specific methodology are taken into account [12].

The Discrete Cosine Transform (DCT) method is commonly used to compress data on images and videos. This process quantizes the frequency of the data and embeds confidential information in the coefficients. However, this may result in images that are sometimes black and white with limited capacity. The method transforms the values of a pixel in spatial domains into coefficients of the frequency domain. Depending on the sub-method used, the image quality may decrease, leading to visual distortion and detectability due to the hidden data [13].

The Discrete Wavelet Transform in numerical analysis is a wavelength transform that simplifies waves, capturing both frequency and time information [14]. The Discrete Fourier Transform method uses a prime even function without complex numbers for statistics and signaling processing [15]. **Figure 6** shows where this


Steganography

Spatial Domain

Frequency Domain

- Least Significant Bits (LSB)
- Pixel Value Differencing (PVD)
- Pixel Indicator

- Discrete Cosine Transform (DCT)
- Discrete Wavelet Transform (DWT)
- Discrete Fourier Transform (DFT)
- Adaptative Steganography

Frequency Domain

- Signals to Frequency
- Math Functions

Spatial Domain

- Image Space
- Pixels

Pixel Intensity

Figure 6.
Frequency vs. spatial domain approach and techniques.

technique belongs among other methods. It involves converting a finite number of equally spaced samples of a function of ordered frequency.

The Adaptive Steganography method utilizes the Human Visual System (HVS) [16]. HVS aims to protect data within pixels less noticeable to humans [17]. Multiple Steganographic methods can be combined to create a sophisticated algorithm that goes unnoticed by humans while embedding secret data into stego images [18]. This method is a mixture of techniques where DCT/LSB applies. It is a combination of DCT and LSB, but it takes into consideration statistical global features. We explored Spatial and Frequency Domains to enhance our method.

The Model-Based method (MB1) embeds information in specific blocks of the image, but it can be easily detected. In contrast, the Block Complexity Data Embedding method (ABCDE) uses watermarks and embeds information at edges. Another approach involves using areas of noise and studying binary patterns that are ignored by the human eye, such as specific colors of the image, like blue [19]. The confidential data is distributed using more than one image and a secret key that is exchanged between the communicating parties [20]. The technique used to hide security data involves splitting it into multiple images, which increases the capacity to hide the data while reducing the footprint. This method aims to prevent unauthorized parties from intercepting the content, as companies must protect their information assets and intellectual property from security breaches. Keeping sensitive data safe is crucial for maintaining a good reputation and avoiding cybercriminals who may exploit vulnerabilities in the device's operating system to view and gather sensitive information. The Frequency Domain Technique is the most reliable framework used for this purpose. However, other commonly used techniques result in a significant increase in image size and distortion when data is hidden, limiting their capacity and increasing detectability.

2.1 The DSoBMP research impact method proposed

Research identifies three major weaknesses of current steganographic algorithms, namely low capacity, high detectability, and distortion. To address these weaknesses and improve data security, a new technique is proposed that focuses on enhancing the capacity of steganography by using multiple carriers. This ensures better protection of data, reduces detectability by embedding data among different images instead of just one, distributes data evenly, and minimizes distortion. The impact of this technique is higher security and protection of steganography algorithms that are used for purposes like copyrights and database security.

The following topics cover essentials that complement the analysis evaluation of current Steganography exploits in this report to prototype the new DSoBMP framework and to test different techniques from current papers using methodologies such as AC Coefficients, ACDCT, Adobe JavaScript, Adobe PostScript, Adobe XML, Metadata Analysis, TJ Spaces Analysis, Arithmetic behind Techniques, Compression for easy hiding, Cover Generation, Cover Stego Attack, Distortion Dynamic-Analysis and Detectable Pattern, Message – Cover Stego Attack, Message Stego Attack, Predicting and Extraction JavaScript, PRNG, Software's to hide data, Spread Spectrum, Statistical Analysis/Anomaly Detection, Stego only Attack, Structural Analysis, Substitution, Substitution of Bytes using XOR and Transform Domain.

The proposed technique offers several advantages over existing approaches. The DSoBMP framework, which is distributed in nature, increases the capacity and decreases detectability and distortion. Additionally, the technique uses the best carrier file to embed data and hides the data using multiple layers of security. This is achieved by using virtually more images originating from the cover medium on the application. The extra partitions on each image are devoted to hiding more information, making it harder to find confidential data since they are randomly distributed in different files. Comparing this technique with LSB alone, which uses one file and any image type, the data is easily found by looking for the least significant bits of the one image for traces of information hidden on LSBs. The main disadvantage is that the technique focuses on BMP, but BMP offers the most capacity increment. However, there is also an improvement in using other image types with distributed steganography. The proposed algorithm is complex and difficult to develop due to the fragments of the file and the data distribution, but complexity in cryptography has always been known to add more security.

2.2 Guide on information about recent work published and gap

The goal of this study was to enhance the security of Steganography techniques through the development of a new framework and algorithm prototype. The purpose of this was to safeguard sensitive data by addressing the weaknesses found in current methodologies. The inspiration for this project came from the identified vulnerabilities of various methods such as Discrete Cosine Transform, which suffer from low capacity, high distortion, and detectability issues and [21] improvements were necessary to address certain issues. Steganography was being used for both positive and negative purposes even before the advent of computers. However, our focus is on using it for the greater good - for protection and security. **Figure 7** explains some of the formal techniques involved in this process. To understand how such attacks listed in the image below can be accomplished, one needs to understand more about computer networks and how the processes of scanning, Accessing and Monitoring can be done in the background with steganography [22].

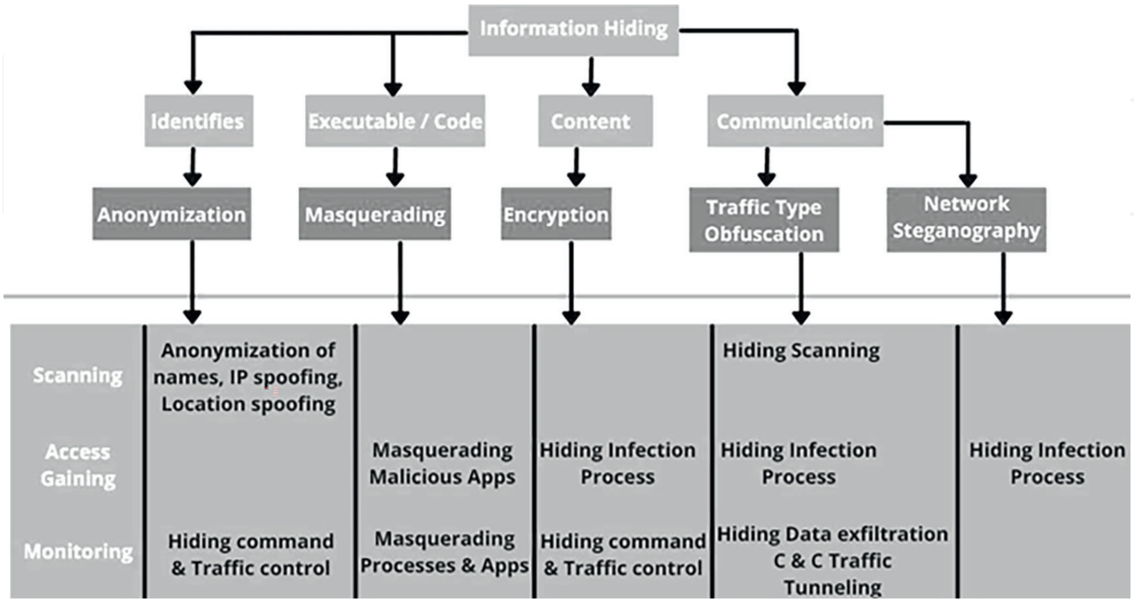


Figure 7.
Formal techniques of steganography.

2.3 Big data, capacity issues and copyright applications

The main task is to address low capacity, high detectability, and distortion in Steganography algorithms. The produced algorithm hides the data among multiple images using BMP as BMP has the purest uncompressed bits that denote higher capacity than other images. By doing that we are maximizing the capacity in two diverse ways, one is by using BMP, which has better capabilities for Steganography, and the second is by using the distributed approach with more images. The structure of BMP is better than other images as the bits are more pure, and uncompressed. The contribution to knowledge is the improvement of the capacity, detectability and distortion using BMP, distributed steganography on Steganographic algorithms. Our research on steganographic algorithms identified weaknesses in popular techniques related to capacity, detectability, and distortion issues. To address these weaknesses, we analyzed existing methodologies and attempted to improve upon them through simulations, measurements, comparisons, and analysis. While many techniques have been used for centuries to secure data, the ever-increasing amount of data requires steganographic algorithms with larger capacity and increased distortion. However, as these methods become more popular, they also become more detectable, necessitating the continuous study and improvement of known methodologies in conjunction with our proposed method.

Cybercriminals put in enormous effort to find the weaknesses of a system, and they release their malicious activities slowly while taking note of how far they can reach. They are also trying to be safe, if not safer than the system, and that is why they succeed. Their success depends on hacking without being noticed, and most of them reach this level quickly. To be secure, we must make sure the data is visible to intended users only, and never let it leak, and if it leaks, we should be able to detect and stop it without damaging the database.

Another point here is the fact that prioritizing the most critical data is essential. In the same way, we restrict employees, for example, by applying access privileges to gain access to some parts of the system as data need to have stringent controls with strong password protection, which is the starting point for protecting the information. It would be great to have 100% security, but it would impact accessibility. The password itself needs to be protected using encryption to hide the passwords from criminals. The algorithm chosen for encrypting the password is crucial. Some algorithms failed on various occasions because of weakness. In later stages, we measure and compare the power of other algorithms including the proposed one as well as other simulations. To provide a good comparison we also consider different files such as different image formats in later exercises that demonstrate the concept of best carriers for Steganographic algorithms in terms of image files.

Some companies still do not encrypt the whole database, instead, they focus on strategy parts to encrypt, like the passwords, and applying other security controls for protection, but any area left unprotected can lead to a security breach. It is essential to perform a backup when dealing with data. If the data leaks and is lost, there must be a plan on how to recover it quickly or have a failover server that will act seemingly. The world today is demanding agile applications and responses, therefore, it is important to dedicate time to risk analysis, risk assessment and recovery plans. Back-up of Big Data has specialized companies that back up in the cloud, mostly a standard for Big Data, these companies also provide disaster recovery to the business these are called Security as a Service, a special area of cloud computing.

Threats are sometimes articulated by people who have intimacy with the system and are otherwise, trusted by the company. It is difficult to find the responsible quickly enough before a data breach. Information Security professionals set up alerts to identify threats, they do identify many indeed if they are watching, but they must act quickly to protect the data while disarming the bomb and then trying to identify the responsible at the same time.

The priority of the Information Security Team in this scenario is to keep all data safe and overcome conflict without damage. We lose control of the data which have leaked, but the data remaining must be protected. If an infected area of the server is affected (the whole server), for example, we might need to take the remaining data out, starting from the most fragile, as in the Crowd Data Scenario, it will take time to transfer everything at once. In the worst-case scenario, transference is a good strategy. If it has a trustful environment, it can be used by whoever is entitled to it and extracted without loss of data control. In the scenario where we leave some data vulnerable as we prioritize to protect sensitive data, we are also taking a risk that can lead to damage, and it would be ideal to have control of where each piece of data is located and get them all safe, controlled and reachable.

It is hard to ensure the owner is tagged to take control of data and ensure ownership. Sometimes this data is lost, and we do not know if the data is used elsewhere, if the new holder applied any security control to override intellectual property, or if the original owner will still be referenced, this is an issue with Crowd-Sourced Data.

Since we are moving to Big Data, and most companies have more than one database to protect, sometimes it is expensively unmanageable for some companies to keep on top of the issue of low capacity and security. Capturing logs is important for identifying issues with behaviors and patterns to analyze data for later working on problem management for investigating eavesdropping, but there are many logs to review, so planning for Big Data is essential as storage capacity is limited and overcrowding a server would slow it down plus it could even crash. An idea is to inbuilt security in each data perhaps using Artificial Intelligence, for location and user interaction logs, but ethical issues must be considered for extracting third-party data and sending information back to the owner, giving automated control and also increasing the amount of memory needed for a system.

Images are copied by thousands of people and used in unusual ways. It is worthwhile having control of intellectual property data logs and data manipulation even as a business idea to monetize the service from the imagined charging peruse and depend on the constraints of different files. Still, we need to consider criminals hacking this model by using screenshots and snipping tools to copy the image, and this can be dealt with by applying law enforcement and making Software have controls on images copied by these means to avoid this intrusive technique.

Moreover, as technology evolves so do hackers. They would attempt to break this system to copy images somehow, and digital Forensics investigators to analyze digital crimes, searching for evidence and presenting proofs of findings, such as how the crime was made possible. It is challenging to crowdsource security and expect only that it becomes more powerful when in fact, sometimes the platform can become more vulnerable as anyone could have access to it and exploit it unless security control and software implementation make use of abstraction methods to customize unique needs and use the trust scenario, where individuals who have contributed in the right way more times get a higher trust certification.

We should not underestimate the fact that ethical hacking, analyzing and mapping paths, and dealing with new threats can lead to better data security, maintenance, and increased security controls. Ethical hacking and Security Audits to find and overcome vulnerabilities ensure data security. Different versions of system bugs fixed increase security, updates play an enormous role in cybersecurity, and using violated software without fixing bugs compromises security.

To protect the data, we need security in-depth, applying security controls and a backup. Capturing a mapping of the data logs to understand where it was visualized in case leakage is necessary, but we must remember to stop snipping Software used to break this pattern to control data copied via applications. The idea of crowdsourcing security could work if individuals were more trustworthy. IT Forensics Examiners should have good technology to identify criminals, the prevention team should be constantly vigilant on alerts and networking monitoring to avoid crimes and capture criminals' tools to protect our confidential or copyright data.

Steganography plays a crucial role in Information Security because it can be applied to all digital files and the issue with privacy and ownership over the web demands awareness of cybercriminals and safe internet usage. Because Steganography will hide the content on the file itself, not requiring the transmission to be highly secure, copyright security applications of digital files such as images, video and audio are one of the main purposes of Steganography algorithms. Without strong security for intellectual property and copyright data, content piracy steals from creative personnel and the industry of software, movies, books music and other content which lose a substantial amount of money to cybercriminals with large-scale copies that are not authorized to circulate on the web, making publishers one of the top clients of securing data using Steganography [23] to embed a hidden message into the archive carrying the digital rights and enabling only authorized people to have certain permissions as in **Figure 8** containing a typical copyright example where images have the details of the owner embedded.

The "invisible" serial number technique and watermarks embed secret data in the files to secure it, banks have started to analyze this approach further to secure confidential data and accounts. The consumer or criminal might not even be aware that these security measures are embedded, and these secret measures can also be used to collect logs which are so important to troubleshoot eavesdropping on attacks over some time as opposed to an attack that is accomplished in a single day.

Encryption can also be used, but the industry finds it beneficial to protect content by concealing it with Steganography for copyrights as unnoticeable marks to avoid

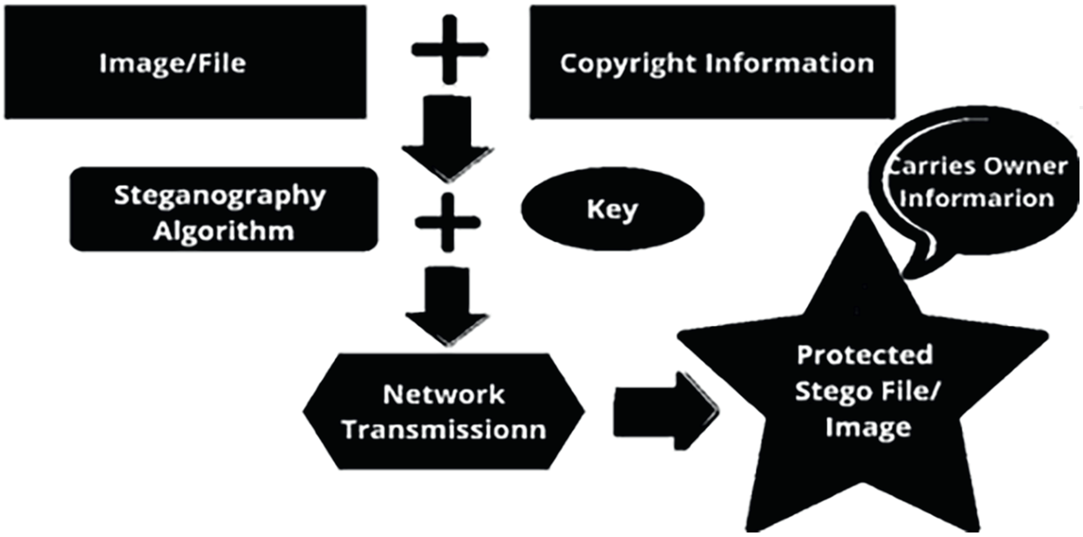


Figure 8.
Typical copyright example.

copies and unauthorized distribution with techniques such as discrete technique per example, to pass as non-existent for the "hacker" [24]. While watermarks denote ownership to the author or company that uses the software, they can also indicate if a document was compromised as different watermark techniques are set to disappear if the document is changed known as the fragile marking method or they are attached to the document in a way that removing would make the data useless.

If a watermark is added to the whole document, the only way to remove it would be by deleting part of the document, hence it could become unreadable as software and if part of the software is deleted to remove the mark, the software would not work as expected or it would be missing features. This method is known as robust marking. Using, the fingerprint method would let the company know exactly who has attempted to modify or distribute the content of an application or a file illegally [25].

2.3.1 Considerations prior DSoBMP

Distributed Steganography is a newer Method using the distribution of data on file(s) and a mix of methods and algorithms. The INFOSEC Convention gave us a few ideas of contemporary issues real companies are dealing with. E.g., the damage of downtime and how attacks that only a few thirty minutes can have a significant impact on the company and if the host companies are at risk, so are we [26]. Focusing on embedding, we researched double embedding and how data extraction does not depend on decryption when reversing data hidden in encrypted images [27].

Also, we studied how vital is Steganography to protect privacy and copyrights, so the original content does not leak or it is lost [28] as well as some crucial terminologies used in Steganographic techniques such as Stego-key that assist in controlling access or recovering the data for people that are aware of its existence, it simply means password and Steganos which is a Stego data containing a hidden message [29] and the converted channel which is the channel of the communication between the sender and the receiver. This is not just used for the transferal of the information data, but it adds extra purposes. We gathered the importance of Steganalysis Software to detect malicious applications and attempt to extract data embedded in the Software to study further to embed data into data [30]. We analyzed how to encrypt the data embedded, so the data is safer inside the file. E.g., embedding the data and then encrypting or encrypting and then embedding [31].

We initiated the research concentrating on the best techniques to embed data so it would be not detected and not extracted or decrypted as one of the definitions of embedding is to make it invisible and undetectable [32], making it a true steganography approach. Plus, if we wish to decrypt, and detect, we should know the best tools and framework for data extraction [33]. We started by researching several techniques such as Cover Generation, Distortion, Spread Spectrum, Statistical, Substitution and Transform Domain, to embed data into an image. We also studied various types of attacks that can be used such as Cover-Stego Attacks, Message-Cover Stego Attacks, Message-Stego Attacks and Stego-only Attacks. We focused on Frequency Domain techniques since it has been the most reliable framework for embedding data into an image. Spatial Domain techniques are not as strong and can be easily detected by Steganalysis [34]. After testing various algorithms, we observed that image size increases when data is hidden. Consequently, we looked into methods of hiding data without increasing size, we were able to simulate all algorithms using one program to test the different algorithms and experiment with different file types. We have different languages from which we simulated techniques, e.g., Python, which seemed incredibly attractive as it is a faster development. The result would be the same independent of the language used, like Java, and JavaScript as the emphasis is always the proof of concept and functionality as opposed to design and beauty. The main task during the experimental phase was digitizing, simulating and testing existing algorithms. By doing that, it was possible to emphasize the creation of a new one with enhancements.

2.4 Protecting copyright with enhanced steganography on images

This method is used to understand how to combat eavesdropping using Steganography, enhancing it to use as a protection technique analyzing studies on both eavesdropping and Steganography defense and how to add copyrights. After understanding how device infection happens with practical experiments, the details gathered reinforce how to identify the steganography practice, highlighting weaknesses in identifying eavesdropping on the device, so we can finalize by developing ways of preventing this interception and enhancing the technique.

We make sure that the developed measure works on mobile devices in general and not just one type of Operating System hence PDFs and Image Files are an excellent approach to follow. This path also takes into account how to use Steganography to combat eavesdropping and protect copyright materials; therefore, this new path involves researching current methods to improve Steganography and use it to benefit security. For this reason, the initial investigation involved eavesdropping and simulating Steganography techniques concurrently to prevent it from being used maliciously not just on phones, but on all devices and also enhancing it for security purposes.

To protect personal photos and copyright materials, using the DSoBMP framework prototype to enhance security, one approach is to add one Steganographic image file that later can be added to a PDF or other document or software with the details of the owner/software to protect a database system, DVD, CD, and any other digital material. To secure and separate the information from the original file, a few carrier images are created from the provided image file, diverting the attention of any interceptor from the actual carriers. The original image is believed to have a stego-image, but its transformed parts contain the data to be protected. This technique can increase the storage capacity by using fragmented images to embed the data, making it a better method for carrying more important data like databases.

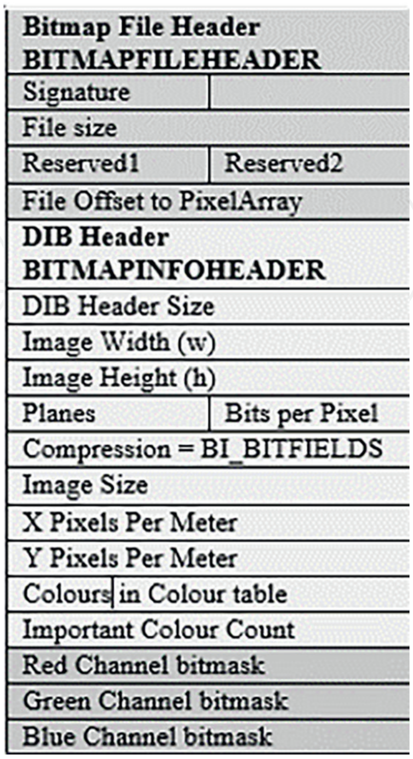
The technique also reduces the chances of detection by partitioning the image into different layers and generating a new set of images that are used as the carrier for the content. This prototype methodology named DSoBMP-I (Distributed Steganography over BMP phase I) application has enhanced the security by decreasing detectability perception and increasing the capacity of the original carrier. It has been found that BMP provides the best security and capacity compared to other file formats such as PNG and JPEG. After conducting various tests, we found that BMP was the most reliable format for enhancing capacity and security. We will discuss the results of these tests in detail in the next chapter. We also applied encryption to the hidden data to add an extra layer of security. We analyzed different encryption techniques and found the best approach for our chosen file format. The Frequency Domain technique was the first technique we analyzed as it is the most commonly used. We also looked at the Discrete Cosine Transform (DCT) method for vulnerabilities and found room for improvement. We included round-off error checks when converting to this format.

The BMP file format is a type of image file that contains a map of bits stored as an array of bytes. Unlike other image formats like JPEG, BMP does not use compression.

Instead, it stores all the bytes of the image. Compression techniques like JPEG are used to reduce the file size of digital images. To work with BMP files effectively, it is crucial to have a good understanding of their structure. BMP stands for "bitmap," and refers to an image format where each pixel is represented by a single binary digit. These files are not compressed, and they provide a way to identify the color depth of an image. Some variations of BMP use different types of compression. The BMP file structure starts with a 14-byte header that describes the file, then the header DIB that provides further information about the bitmap and its pixel format, as illustrated in **Figure 9**.

The compressed version of BMP comes with some optional functions like the Extra Bitmasks. An optional color table is also included in this structure, which is followed by the Gap1 block that defines the alignment structure of the file. The Pixel array structure is mandatory and present in every BMP file, which specifies the value of each pixel. It varies in size. The next two structures, Gap2 and ICC color profiles, are optional and vary in size. They help manage the colors [35]. Our algorithm has shown that the BMP format produces the best results in terms of detectability and size increase after embedding. To demonstrate this, we conducted an experiment where we embedded a stego message of 4096 bytes into various file formats of the same image. The purpose of this experiment was to compare detectability while taking into consideration the size increase after embedding. The following text displays the outcome of this experiment.

The BMP file format has optional functions such as the Extra Bitmasks, which can be used with the compressed version of BMP. Additionally, a color table is included in this optional structure along with the color pixel array and the Gap1 block, which defines the alignment structure of the file. The pixel array structure is mandatory in every BMP file to determine the specific value of each pixel, and its size varies. The next two structures are optional. In this report, we present the results of an experiment where a stego message of 4.096 bytes was embedded into different file formats



Bitmap File Header **BITMAPFILEHEADER**	
Signature	
File size	
Reserved1	Reserved2
File Offset to PixelArray	
DIB Header **BITMAPINFOHEADER**	
DIB Header Size	
Image Width (w)	
Image Height (h)	
Planes	Bits per Pixel
Compression = BI_BITFIELDS	
Image Size	
X Pixels Per Meter	
Y Pixels Per Meter	
Colours in Colour table	
Important Colour Count	
Red Channel bitmask	
Green Channel bitmask	
Blue Channel bitmask	

Figure 9.
A BMP file structure.

of the same image. The objective was to compare detectability while considering the size increase after embedding the message. We began by analyzing the exploitation of image formats and frequency domain techniques, providing a quick introduction to these topics. The simplest technique used for embedding data inside the file is exploiting areas where usually no data is kept. This technique does not alter the content of the file but is visible when looking at the source code and placeholder. BMP has shown the best results in several aspects.

The report discusses various methods of applied steganography, including simpler methods such as EOF and EXIF. However, the report focuses on a more advanced technique that combines Discrete Cosine Transform with well-known technologies to enhance steganography and improve copyright protection. Discrete Cosine functions were used as they are the best-known mathematical periodic functions. Fourier series were also employed to analyze a periodic function into its constituent components and send signals without distortion. The technique involves breaking images into sub-bands, deleting high-frequency components and using only real numbers for JPEG compression. **Figure 10** illustrates the standard DCT algorithm that contains LSB, as shown in **Figures 11** and **12**. When embedding, the images are broken into four

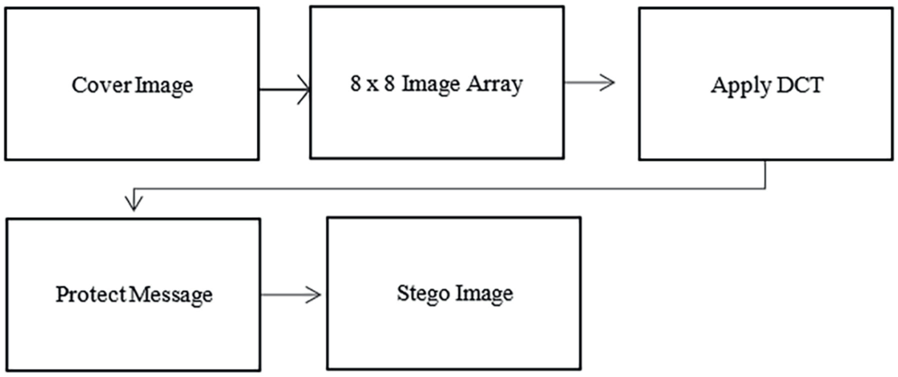


Figure 10.
A basic DCT algorithm.

Figure 11.
8 × 8 base matrix of DCT.

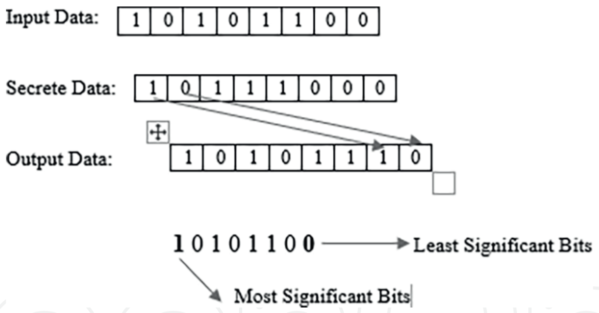


Figure 12.
LSB and MSB insertion example.

steps before applying DCT and genetic algorithms. The Cosine Transform algorithm code simulates breaking the figure into partitions of an 8x8 block, which is then reassembled.

Quantization is a signal processing technique that involves rounding and truncating a matrix setup. Matrix multiplication is then performed and the result is stored in a temporary matrix N × N.

This procedure involves using cosine to re-multiply and generate the final matrix. The value is then quantized and rounded to an integer. The inverse operation is the next step, which outputs 0 to 255 values for the pixels in an N*N matrix. **Figure 12** above illustrates the concept of Least Significant Bits (LSB) and Most Significant Bits (MSB) of an image array. DCT incorporates LSB as part of the technique, in addition to other measures it includes for confusion and diffusion. This adds more complexity to the distribution of secret data, strengthening the security of the algorithm beyond just using LSB alone. Every pixel has 24 bits, to modify a pixel programmatically would depend on the algorithm used, but it is easier to use a sequence. The pixels to be modified must be selected randomly, using pseudo-random, and the algorithm needs to remember which pixel is picked. Examples of algorithms' methods to change the file and embed content would include a region of the file to be replaced, for example, edges, highest frequency, and least significant bits. There is a choice to pick one bit in a byte or more. To minimize the number of modified pixels, we primarily employ the LSB algorithm (Least Significant Bits), and the number of altered bits per pixel is directly linked to the Data rate. When we use algorithms that try to concentrate all the confidential data in as few pixels as possible, we are more likely to get visual distortion and be more vulnerable to statistical analysis, as on the algorithms we simulated previously. Hence, modifying as few pixels as possible will make data contained in one place which will be discoverable easier and be more vulnerable to statistical analysis. Some of the key issues are illustrated in **Figure 12**.

The process of hiding data in an image involves compressing the image, which in turn leads to increased distortion as the image quality decreases. It is crucial to note that this method is sensitive to statistical analysis, and using specific bits to hide data can make it easily detectable. Some algorithms only output black or black-and-white images after data is hidden. Furthermore, the capacity to hide data is limited, as it can only hide data in specific bits of an image, such as LSB, HSB, and specific colors.

Detecting steganography can be done by observing distortion in the file. However, visible distortion is not always a reliable indicator. Mathematical algorithms can help identify hidden content, but some algorithms may have minimal distortion and still be detectable by statistical analysis. It's important to note that some images may be distorted due to resizing, and not because of steganography. For example, they are

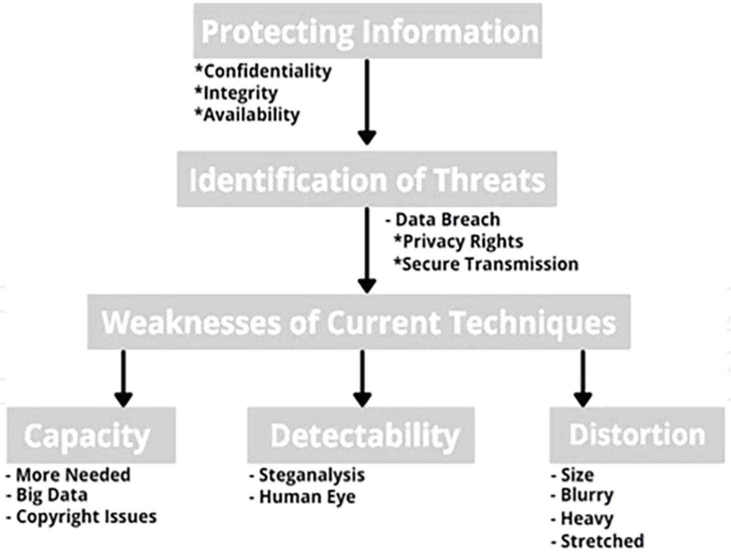


Figure 13.
Issues of DCT as a steganography technique.

stretching an image too much by maximizing or minimizing it. One algorithm that may not always show distortion is LSB. The content hidden by LSB is located in the least significant bits of the carrier file. Mathematical algorithms can be applied to identify the LSB capability.

Different algorithms have different capacities for measuring hidden content. The least significant bits are often the easiest targets for steganalysis. Other algorithms, such as those that search for specific colors, can also restrict capacity. **Figure 13** provides more examples. The Discrete Cosine Transform method algorithms are widely used in the Steganography Transform Domain, but there are still weaknesses that need improvement. The capacity, distortion, and detectability of DCT depend on the amount of data on the carrier file. The coefficient bits in the transform domain of the file are used to hide content in the stego-image.

For instance, let us consider an 8 x 8 block per RGB (Red, Green, and Blue). This converts into 64 coefficients that undergo DCT compression and quantisation. Additionally, LSB is used to substitute the image coefficients with the secret content. However, there is a need for more capacity with minimal distortion and detectability so that more data can be hidden securely. Creating a superior quality stego-image that can be measured by dB is the ultimate goal.

Therefore, we simulated and tested improvements for our new algorithm using distributed Steganography and a mix of methods that worked well to enhance Steganography itself. Our analysis of different digital data files helped us determine the best one to use for securing content while improving capacity and minimizing detectability and distortion.

3. Conclusion

The main weaknesses of current Big Data techniques have been identified, and important points for analysis and investigation have been highlighted. The DSoBMP-I approach can address the low capacity, high detectability, and distortion issues that are common in most steganographic algorithms used today, particularly DCT. After experimenting with different ideas and image formats, it was found that BMP is the best image type for embedding data. To ensure better flexibility, two useful encryption algorithms were demonstrated in case the embedded data or files are large and complex, and extra security is needed. Additionally, a new method of distributing personal data into a set of images was proposed, based on the original file supplied, along with the choice of partition size.

This approach has been shown to increase capacity by 100% when compared to other algorithms that do not consider file format as a crucial factor and do not have access to distributed Steganography. It achieves this by using a simple 2x2 matrix, and the capacity further increases as the matrix size increases. This approach has been proven to be more powerful than previous research, such as Nidhi's study mentioned in the text. The algorithm used in this research has proven to be superior to previous methods. The result of 68db of power achieved in this method surpasses the results of other research that highlighted the power of 30db in 2010 and 65db in 2015, with similar experiments. This is a significant step forward. The detectability and distortion are not easily noticeable, and hence, we recommend using the DSoBMP-I methodology proposed here for securing copyright materials and big data.

The improved security of the steganography algorithm is attributed to two factors: a larger spatial domain for embedding the secret data and the use of digital data files with cleaner pixels. By distributing the data across more files, we increase the area available for embedding, which in turn results in less distortion and less detectability due to random allocation. Furthermore, encryption in a larger domain using the most efficient techniques was tested during this research and compared with other approaches and files. Practitioners already using steganography today will benefit from extra security and less distortion and detectability of their copyright data on their images, they will also benefit from embedding more details on their copyright note, license terms, history of the file or metadata of the image, more data can be recorded and kept safe. In summary, improving capacity provided the largest set of significant clusters for this investigation or in other words having a bigger space to embed data, also improves the security of the secret message as it will be distributed in a bigger domain hence it will also have less detectability and distortion.

The uniqueness of the new algorithm and our published papers cited already by other researchers are proof of the contribution to knowledge. In this report, we have a clear guide of historical and current techniques in steganography, apart from highlighting where criminals can embed secret information on PDFs for investigators to analyze, the author identified the best file to use to embed data to secure copyrights and tackled weaknesses of Steganography with a new algorithm to enhance the security of the technique.

There is always room for improvement in any project, and this research is no different. With this research topic, the author was able to gain knowledge on how Steganography can be used in diverse ways and different technologies and how it still needs to grow to be able to survive big data and the cybercriminals that make use of it. It was remarkably interesting to know exactly where data can be hidden to follow up on further Steganalysis projects and Steganography itself, used for thousands of years, sometimes people might think that it could be dying, but it is not dying, it is growing stronger so that it might need to be better regulated in the future. Other researchers can start their work straight away after learning the best file to embed data from this research, the weaknesses of current techniques, how to address the issues and considerations for extra security for applications and improvements suggested here.

We aimed to protect copyright data and improve security, but improving security is an ongoing task as digital criminals are always inventing new ways of breaking algorithms to achieve their targets. The strongest part of the work is the good analysis and contents on several different methods shown here and using different data files to show and simulate steganography in various forms using different techniques, while the weakest part is attributed to the design of the prototype, the author wanted to create a web application to add copyright to images and ensure they all carry the data across the web to identify the owners and creators, but it is rather disappointing the number of obstacles behind the international medium Internet, how we regulate the Internet to make this happen is a question that needs to be answered before this is made possible for the public and with big data, snipping tools and other applications that can copy images.

The most interesting findings and recommendations are how data can be manipulated in different ways and embedded inside different files, masked, and compressed and how criminals even with the technology we have today can still distribute a virus over the internet without leaving any traces, the more technology evolves, the more cybercriminals evolve, and how difficult it is for us to have these "traces" and records from the origin of files, finding the real owners of the data over the web because the internet is not regulated well. The author has always been in favor of freedom of expression on the web, but after investigating steganography and eavesdropping, we need to think of ways of regulating and attributing ownership to files and information circulating on the internet. For the proposed further research, the author would start by concentrating on legal issues of the internet medium, researching on providing guidelines to regulate it better or part of it to be able to identify ways of making sure we can cope with Big Data in terms of attributing ownership to internet users and creative personnel, authors or creators, definitely embedding precise location from where images are created, then carrying logs of where and who has accessed specific materials from specific sites. This would be extremely challenging as the internet is an international medium regulated by different laws from different regions and because of the number of logs that had to be recorded by someone in an exceptionally large server which would be expensive, but for easier starts, this research can also progress by improving the design of the current solution.

Extra considerations with eavesdropping and snipping image tools, embedding data via the WIFI can also add security as data can leak while being embedded plus the research for the new smart data traveling the network with embedded characteristics from the owner is complex but still worth researching more. The improvements never stop, and we should consider protecting the data while embedding it on the network and limiting the clients that "purchase" the software.

In the next session of this chapter, we will encounter the references used to accomplish this research. Published work in international journals such as Springer includes experiments on PDF by highlighting exactly where steganography can be added to identify malicious intents, as well as security awareness for eavesdropping data from mobile devices and the improved algorithm to use steganography to tackle weaknesses of current techniques to secure copyright materials like images from smartphones of creators as an example, but from the process of doing a PhD, the author learnt, apart from the knowledge in this field, that there will not be a time when we have all the answers and we will know everything, technology is always changing, the more we read, the more we learn, and the more questions we have, and we can always research more and as we find fixes for a problem, another one arises, and we can always learn more. The advice the author would give to researchers entering this area is to follow the recommendations of this chapter for a straightforward start on your research and focus on one problem at a time, as in real life, the issues and work to strengthen security never ends.

Acknowledgements

A big thanks to my beloved mother Veronica Maria de Araujo who always persuaded me to complete my doctorate and kept me motivated to continue the research, never stop searching for answers and believing in Science. Another big thanks goes to my husband Adriano Ribeiro da Cruz for always being present in all hard and good moments throughout my academic life facilitating the completion of research work. In addition, my rescued dog Mickle Pickle, my black ex-racer greyhound provided joy and mental stability through the kindness in her eyes, smile and company for exercise walks during her retirement.

Author details

Istteffanny Isloure Araujo
Intelligent Systems Research Group, School of Computing and Digital Media, London Metropolitan University, London, UK

*Address all correspondence to: i.araujo@londonmet.ac.uk

References

[1] Kalaivanan SA. A survey on digital image steganography. International Journal of Emerging Trends and Technology in Computer Science. 2015;**17**:30-33

[2] Zhang R, Dong S, Liu J. Invisible steganography via generative adversarial networks. Multimedia Tools and Applications. 2018;**78**:8559-8575. DOI: 10.1007/s11042-018-6951-z

[3] Koptyra K, Ogiela M. Distributed steganography in PDF files—Secrets hidden in modified pages. Entropy. 2020;**22**(6):30-59

[4] Mills R. The Metadata in JPEG files. 2018. Available from: https://dev.exiv2.org/projects/exiv2/wiki/The_Metadata_in_JPEG_files [Accessed: May 15, 2021]

[5] Chandramoulia R, Memon N. Steganography Capacity: A Steganalysis Perspective. Vol. 1(1). New York, USA: A Department of E.C.E., Stevens Institute of Technology; 2015. pp. 1-5

[6] Kawaguchi E. Applications of Steganography. 2015. Available from: http://datahide.org/BPCSe/applications-e.html

[7] Jahankhani H et al. Conference proceedings. In: Jahankhani H et al., editors. Global E-Security. London: Springer; 2011. pp. 23-25

[8] Siper A et al. The rise of steganography. Proceedings of Student/Faculty Research Day. 2005;**1**(1):1

[9] Korus P, Białas J, Dziech A. Multimedia Tools and Applications. 2014;**68**:59. DOI: 10.1007/s11042-011-0986-8

[10] Mondal B, Mandat T. A Secret Shearing Algorithm based on LSB Substitution. 2014. Available from: https://www.researchgate.net/figure/A-typical-diagram-of-LSB-Substitution-techniques_fig1_262996420 [Accessed: May 15, 2021]

[11] Borse D, Patil S. Review on transform domain steganographic techniques (DCT and DWT). International Journal of Innovative Research in Computer and Communication Engineering. 2015;**3**(12):12466-12473

[12] Thampi SM. Information hiding techniques: A tutorial review. LBS College of Engineering. 2007;**1**:1-15

[13] Dave HP. Steganography technique based on DCT coefficients. International Journal of Engineering Research and Applications. 2012;**2**:713-717

[14] Elham Ghasemi JS. High-capacity image steganography using wavelet transform and genetic algorithm. Proceeding of the International Multiconference of Engineers and Computer Scientists. 2011;**1**:495-498

[15] Manda N. Image authentication technique in frequency domain based on discrete Fourier. Proceedings of ICCS. 2010;**125**:144-147

[16] Mazurczyk W, Karaś M, Szczypiorski K, et al. YouSkyde: Information hiding for skype video traffic. Multimedia Tools and Applications. 2016;**75**:13521

[17] Rana S, Sur A. View invariant DIBR-3D image watermarking using DT-CWT. Multimedia Tools and Applications. 2018;**79**:1-29. DOI: 10.1007/s11042-018-7024-z

[18] Tataru R et al. Is hidden data safe? Analysis of the public

cryptand hide-steganos application. Proceedings of the Romanian Academy. 2015;**16**(1):299-312

[19] Leiner B, et al. Brief History of the Internet. 2017. Available from: http://www.internetsociety.org/internet/what-internet/history-internet/brief-history-internet

[20] Zielińska E et al. Development trends in steganography. Institute of Telecommunications. 2015;**1**(1):1-13

[21] Antony N. Improved capacity collage steganography using discrete cosine transformation. International Journal of Scientific and Engineering Research. 2015;**6**(11):1060-1064

[22] Cameron L. With Cryptography Easier to Detect, Cybercriminals Now Hide Malware in Plain Sight. Call It Steganography. Here's How It Works. 2018. Available from: https://publications.computer.org/computer-magazine/2018/11/15/how-steganography-works/ [Accessed: May 16, 2021]

[23] Evsutin O, Melman A, Meshcheryakov O. Digital steganography and watermarking for digital images: A review of current research directions. IEEE. 2020;**8**(1):166589-166611

[24] Al-Thahab O, Hussein A. Implementation of stego-watermarking technique by encryption image based on turbo code for copyright application. IEEE. 2020;**1**(1):148-153

[25] Garg M, Gupta S, Khatri P. Fingerprint watermarking and steganography for ATM transactions using LSB-RSA and 3-DWT algorithm. IEEE. 2015;**48**(1):246-251

[26] Checkpoint. Infosec 2015 - Data Center Security. Infosec 2015- Data Center Security, Check Point. London: Tele Group Ltd.; 2015

[27] Sanyal T. Reversible and Irreversible Data Hiding Technique. Hyderabad, India: Neudesic India Pvt. Limited; 2014. pp. 1-4

[28] Barlow J. Chapter 20. In: Barlow J, editor. Copyright and Privacy Protection. Cambridge: University of Cambridge; 2014. pp. 424-452

[29] Neil Johnson Z. Information Hiding: Steganography and Watermarking-Attacks and Countermeasures. New York: Springer Science + Business; 2012

[30] Shankdhar P. Best Tools to Perform Steganography. Wisconsin, USA: Infosec Institute; 2015. pp. 1-10

[31] Bhure SB. Data encryption by image steganography. International Journal of Information and Computation Technology. 2014;**4**:453-458

[32] Sadhana G. Strengthening the security of information use. International Journal of Computer Science and Information Technology Research. 2014;**2**:27-35

[33] Poretsky S. How to decrypt messages embedded within images. CHron. 2016;**143**:1-5

[34] Kalaivanan SA. A survey on digital image steganography. International Journal of Emerging Trends and Technology in Computer Science. 2016;**1**:30-33

[35] Ramapriya B. An improved approach of text steganography in application with rotational symmetry. International Journal of Innovative Research in Computer and Communication Engineering. 2017;**5**(7):12939-12947