**Improving Steganographic Capacity Using Distributed Steganography over BMP**

**Istteffanny Isloure Araujo · Hassan Kazemian**

**Intelligent Systems Research Centre from the School of Computing and Digital Media of London Metropolitan University**

## **Abstract** Our research area tackles the improvement of private data security using our proposed Steganographic method called DSoBMP-I (Distributed Steganography over BMP faze I) to improve the issues of low capacity, high detectability and distortion. The methodology consists of a new distributed steganographic approach to minimise the main weaknesses of today’s methods, including Discrete Cosine Transform, where the capacity, detectability and distortion need an upgrade to accommodate secure steganography for our data protection. The proposed prototype approach that evolved after a few experiments using our distributed steganographic method, where secret data is secured into a set of BMP files (as it is proven more reliable), originates from a raw file that is not necessarily a BMP at the start. After applying a layer of encryption for extra security using two different methods such as RC4 & RSA, comparing the two encryption techniques for their agility and extra security to address the issue of low capacity and better security using the DSoBMP-I method, all deriving from the supplied image. The overall achievement was improved capacity that doubles as the set of BMP images increases, less distortion and detectability as secret data stays among different files.

**Keywords**: BMP, Cryptography, DCT, Encryption, Security, Steganography.

# **1 Introduction**

Steganography is the act of hiding information to protect media database systems, digital content access control, data alteration protection, confidential communication and secret storage within text, video, audio and image files. Steganography consists of hiding some content inside another while steganalysis is the process of identifying steganography itself. It is important to note that steganalysis does not reveal the hidden message identified [13]. To uncover the secret message, Cryptography is the method of breaking mathematical algorithms within secret communications. This study aimed to improve the security of Steganography techniques by examining and combining current methodologies to extend them to a novel approach to protect sensitive data. The inspiration comes from vulnerabilities of different methods such as Discrete Cosine Transform identified with low capacity, exposed to distortion and detectability issues [2] hence improvements became necessary to address and have been extended to all other current techniques. The research motivation is to tackle current vulnerabilities of Steganography algorithms and use the improved technique to protect data such as copyright materials. The principal vulnerabilities are Capacity, Detectability and Distortion; we address all of them by improving the capacity with distributed Steganographyy that also makes the protected data less detectable and less distorted. The importance of fixing this problem is the fact that copyright materials, for example, are secured in a better manner, and we can avoid infringement to copyright in this example; however, the same technique may apply to protect secure communication, database systems, digital systems and access control. The functional impact of this new approach is a significant improvement in the security of the algorithm to protect personal and confidential data. The main advantage of this technique compared to existing ones is the capacity improvement to protect the data on images as for the copyright example, detectability and distortion also had improvements. There are three main weaknesses in current Steganography algorithms, being they low capacity, high detectability and distortion. This research is relevant because it addresses all these three weaknesses, improving data security. The technique presented here concentrates on improving the capacity of Steganographyy by using multiple carriers; hence data is better protected, the detectability becomes less noticeable as the data is embedded among different images, not just in one and as the data is well-distributed distortion is less likely. In the real world, the impact of this technique is higher security and protection on Steganography algorithms used for Copyrights purposes and Database Security for example.

The main pro is that this technique increases capacity and decreases detectability and distortion compared to existing approaches. It hides the data using more layers, increasing security as the number of partitions on each image is denoted to hide the information. 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 possible information hidden. With this technique, it is harder to find the secret data since shared on different files. The main disadvantage is the focus on BMP. BMP makes the most of the capacity increment, but there is also an improvement using other image types with distributed Steganography. Different experiments were carried starting with PDFs as they are the most commonly available, passing to Word, Excel, Video and then images that incorporate any of these files via embedding messages on pictures via the most common method the “Least significant bits”, where some bits are substituted. Steganography succeeded for many years since its first historic record dated back to 440 BC, and its importance among all people who incorporated the method over the years even before computers even existed shows highly.

During 60 BC, the Egyptians had a simple cypher solution named after Julius Ceaser, today, known as Ceaser Cypher, consisting of interchanging alphabet letters [3]. In the 5th century, the Greeks exploited steganography with wood panels having messages camouflaged with wax, and later with slaves with heads with hidden messages covered by grown hair [4]. In the 15th Century, Trithemmius, a German rector, produced a piece of work named Steganographia related to magic, containing cryptography and encryption discourses [3]. Germans had photos hidden on documents using tiny dots that later were read by spies with magnifiers who interpreted each photo symbol as a pre-defined warning during the World War between 1939 and 1945. Encoded messages have been knitted into sweaters using Morse code. Invisible ink was exploited by the Americans where messages were written mainly with milk, vinegar, juice and even urine [5]. In 1961, the internet became popular, and individuals started to confront digital crimes and illegal activities over the web. The first attempt to secure our information was Cryptography [3], a security technique used before computers to protect several types of information. Within Cryptography, this specific method evolved from the Ancient Greeks to today’s modern society: “Steganography”[14], merely meaning “information hiding”, continues to be used to secret camouflage messages during conflicts [21, 17]. 1968 was the year; pictures were taken of American prisoners by the North Korean military to demonstrate the soldiers were alive and well. They used finger gestures that Americans decrypted when receiving the photos [3] while in 1998, some governments adopted steganography watermarking to secure intellectual property like music, multimedia and other copyright materials [7].

In 2001, Steganography over the web had good and bad usage; therefore, the scanning of two million eBay pictures took place on suspicion of criminals using the site for other than selling [8]. People realised the importance of continuing to study steganography and steganalysis as it can carry not only private secrete legal data that needs to be protected, but it can also carry child pornography, and terrorism-related messages to avoid. Hence, Steganography became a popular research area, and several papers started to emerge with new techniques to protect our data and to identify lousy content to hide. [Krzysztof Szczypiorski](https://en.wikipedia.org/w/index.php?title=Krzysztof_Szczypiorski&action=edit&redlink=1) removed some of his research from the internet. After that in 2015, Mazurczyk et al. published the first book on Network steganography [3], despite the criminalisation by the US government as not all countries have taken the same approach and the primary purpose is always the security of our data. Image Steganography is the spotlight of this research as per the possibility of adding into most types of files like PDF, Word and Excel. There are different methods of embedding content on images here. There are several different ways of securing data within a file conveniently without complication; modifying the file structure is one of the most straightforward ways to do it without much effort. The EOF and EXIF methods are examples of applied Pure Steganography. The EOF method adds a message to the End of the File object in the code [10], where the compiler ignores the data and does not display it in plain sight when the file is opened [30]. The EXIF method adds a message to the description of the file using a special camera or image organiser software, nothing is displayed itself when opened, but instead, it displays as a description of the file [16], and it can only show with the help of the software used to protect the data. These methods modify the size of the file, as more characters are added to it [12].

Given an image is a pixel matrix, the Spatial domain of every image is the image itself. The Spatial Domain method for Image Steganography, though, is called a Spatial Domain technique a method that modifies the values of the pixels of an image, even though this is a reliable method, Spatial Domain techniques detected by steganalysis [39]. It is used as an example of Private Key Steganography, meaning data using a key fed to both ways: embed and extract, but it works without a need for any apparent key fed as some programs are known for generating the key randomly via algorithms on the code. The *LSB methodology* uses less significant bits to attach the data. It is widely available, and one of the most explored methods of the Spatial Domain category [28]. It works with any file format, but the problem is detection by statistical analysis that starts typically by reading these spaces. [17]. The *histogram-based Data Hiding* inserts data on the Highest Frequency bits of the image, increasing robustness. It considers how intense the pixels are within a black and white or colour image from which the RGB value can be measured (red, green and blue) together with its brightness and contrast [4], hiding data in specific colours and intensities of a stego image. The *Frequency Domain* of an image as a Steganographic technique is an enhanced method that processes the carrier image according to its transform. It secures data by altering the image with one of the methods below, and it generally outputs a different image with the hidden data embedded in it. This method considers greyscales, frequency of content and the specific method utilised [32].

The *Discrete Cosine Transform* method compresses data in quantised frequency on images and videos. The secret data is embedded in coefficients, making it black & white with limited capacity, transforming values on the pixel in spatial domains to the coefficient of the frequency domain. It decreases image quality as it hides data [9].

The *Discrete Wavelet Transform* method in numerical analysis, is any wavelength transform in which the wavelet samples discretely. Captures both, frequency & location in time information [10]. The *Discrete Fourier Transform* method uses a prime even function without complex numbers for statistics and signalling processing [20]. The *Adaptive Steganography method* is based on the Human Visual System, mostly referred to as HVS [21]. HVS aims to protect the data within the pixels that are less noted by the human eyes [26]. This technique can combine one or more Steganographic methods to form a complex algorithm that will not be noticed by humans when it outputs a stego image after the secret data is embedded [31]. The *Model-Based method* (MB1) embeds in certain blocks of the image but is easily detected. The Block Complexity-based Data Embedding method (ABCDE) uses watermarks and embeds at edges [11], another approach uses areas of noise and studies of binary patterns that are ignored by the human eye like specific colours of the image such as blue [18]. A secret key is common between individuals involved in the communication. It relies on more than one image to distribute the hidden data [35]. It distributes the data reserved to secure into a set of different images, and for this reason, it benefits from more capacity to hide the data and lower footprints on the result. All methods have the same intent of stopping interception of the content from the unauthorised party, as companies must protect their information assets, and intellectual properties from security breaches and keep sensitive data private to have a good reputation on the market.

The Frequency Domain Technique is the main one used nowadays as it has been the most reliable framework as Spatial Domain techniques are weak when using steganalysis [39]. Kalavainan has written a paper highlighting current techniques available in Image Steganography and the issues. On most used techniques, the image increases when data is hidden and increases distortion, gathering other techniques learnt about limited capacity and detectability. Therefore, the a need to improve the overall quality based on Capacity, Detectability and Distortion. For this, the methodologies already in place were detailed together with other attempts of similar ambitions. Our research and experiments observed and measured how the most popular techniques reacted to it to create a new approach addressing issues with all these known methods which will be described better in Chapter 2.

In Summary, many techniques have been used over centuries to accomplish the security of data, and this data has become bigger and bigger every year, calling for more capacity with steganographic algorithms that can affect distortion. As methods became popular, detectability also did, and the need to work on these weaknesses to strengthen security is an ongoing matter that will never die until steganography lives. **Section 2**, is a study of mainly known methodologies in conjunction with our proposed one. **Section 3** brings several experiments and results that prove our concepts and choices before finally concluding the thesis in **Section 4** and providing valuable references after that.

# **2 Methodology**

After having researched and simulated the known techniques, a new method which is a combination of most methods above is proposed, however mostly inspired by the weaknesses of Discrete Cosine Transform with its capacity, detectability and distortion issues together with the need to improve Steganographic security. The new approach creates a few carrier images from the original supplied image to embed the information, keep it secure and separate from the original file provided, taking the attention of the interceptor away from the actual carriers. The original image is suspected to be the carrier, and may not contain embedded data, but its transformed fragments will not just contain the data that needs to be protected. However, it can find a maximised capacity due to the number of fragmented images generated to embed the content, making it a better carrier for more critical data such as databases and minimising the detectability as hiding content with the new approach of different layers of the partitioned image that generates a set of new images used as the carrier. It has increased the capacity of the original carrier, decreased the detectability perception and enhanced security due to the applied methodology named *DSoBMP-I (Distributed Steganography over BMP phase I)*.

There is a reason why the primary method has focused on BMP to provide the best security and improve capacity. After measuring the effectiveness of BMP, PNG and JPEG, among others to gather the most reliable to increase the gear of capacity and security, BMP came across the best results. Besides, encryption was applied to the hidden data, adding an extra layer of security or agility measuring the best approach to take and analysing the results of different encryption techniques on the chosen file format. One of the first Algorithms tested was the Derek Upham also known as JSteg, and that worked well to hide the data bit by bit overriding the least significant bits. Then, the F4 algorithm where data goes inside the coefficients [22] and F5 embeds via permutation and needs uniform distribution to cause less damage to the image [23]. For the Histogram embedding, which includes the frequency domain technique, adaptive Steganography uses the discrete inverse Fourier (DFT), and it is a statistically aware embedding method also using the least significant bits as opposed to MB1 known as the Model-based method for distribution with minimum distortion [24]. Furthermore, the Frequency Domain technique was the first gathered for further analysis because it is the most used and it encompasses round-off error when converting, and within it, Discrete Cosine Transform (DCT) method vulnerabilities used as inspiration as it has notable room for improvements.

## **2.1 BMP**

The Bitmap (BMP) file format is a file generated from a map of bits, known explicitly in computing as a bit array. Most of the image files are denoted in a similar manner; however, BMP has its unique charm since all bytes are entirely present in the file while other formats use compression from the start. An example is JPEG, and compression applied to digitalised images [6]. It is essential to understand the file structure of BMP to program and explore all means of this format to improve our results. Its structure is initially for the Operating System Windows, and another advantage is that many applications and digital systems support it. When referencing to BMP, it implies one bit to one pixel of the image; these are not compressed and identify the depth colour of a digital image. Other variations of BMP use compressions named differently. BMP file structure starts with a header of 14 bytes describing the file, followed by a DIB header detailing the bitmap and its pixel format as illustrated in **Fig. a**. There are optional functions such as the Extra bitmasks to be used in conjunction with the compressed version of BMP. A colour table is also aggregated into this optional structure bringing the colour pixel array, followed by the Gap1 block defining the alignment structure of the file. The Pixel array structure is not optional, every BMP file will have this to stipulate the specific value of each pixel, and this varies in size. The next two structures are also optional, Gap2 and ICC colour Profile vary in size and help to manage the colours of the file [25]. As BMP has shown the best results in several aspects, our example algorithm described in **Table 2** shows the outcome of one of these experiments where the stego message has 4.096 bytes embedded into different file formats of the same image to compare detectability taking into consideration the size increase after the message is embedded.

## **2.2 Discrete Cosine Transform**

Most Digital Images use Discrete Cosine Transformation for digitalising content and compression; one example is JPEG, DCT is part of its compression and the file format itself. Since we are working with digital images and Steganography where the carrier will take compressed data, DCT comes in handy, and the same approach is sufficient for any other file format to compress and embed information. This study aimed to improve the security of Steganography techniques a novel approach to protect sensitive data. The inspiration comes from vulnerabilities of different methods such as Discrete Cosine Transform identified with low capacity and exposed to distortion and detectability issues [2] hence improvements became necessary to address. The technique starts with the cover image; we partition it into a sub-bands array; small high-frequency components are deleted and use only real numbers for JPEG (Joint Photographic Expert Group) compression [19]. When embedding, images pass into four steps before applying DCT and genetic algorithms. The Cosine Transform algorithm code simulated breaks the figure into partitions of an 8x8 block as an example, and reassembled it [19] as per the steps below:

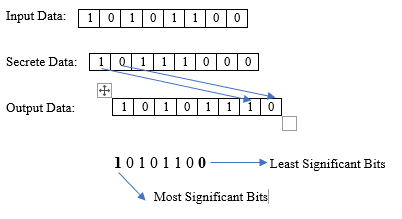
1) DCT Object Created.

2) Use the 8\*8 matrix to compress.

3) Use the dequantitizeImage () method.

4) Use inverseDCT () method.

There is a quantisation (signal processing like rounding and truncating) matrix set-up. Then, matrix multiplication is performed and put in a temporary matrix N \* N which uses cosine to re-multiply to generate the final matrix, quantise and round the value to an integer, the inverse operation takes place, and the N\*N matrix will output 0 to 255 values for the pixels [32]. **Fig. a** shows the concept of Least Significant Bits and Most Significant bits.



**Fig. An** LSB insertion example

### **2.3.1 Limitations Highlights: Capacity, Detectability and Distortion**

Capacity plays a crucial role in the security detectability of steganography algorithms, apart from already being known as the primary measurement for Steganography algorithms [32]. Another measurement used is the Peak Signal-to-ratio noise (PSNR); it outputs decibels (dB). A higher PSNR will denote better security and quality on the result of the algorithm. This measurement relates to the quality and security of an image when modifications like our steganography take place. Some algorithms with higher capacity tend to be weaker in security detectability and, hence not very proficient with steganalysis, e.g. statistical analysis. Some high-capacity algorithms also present more distortion on a steganography file; therefore, the idea of improving capacity with minimal impact on detectability and distortion is an excellent topic for research and contribution to knowledge as they usually have an enormous impact on each other.

1. It increases distortion as quality decreases as it compresses the image to hide the data [25].
2. It is sensitive to Statistical Analysis as per item 4, using specific bits to hide data makes data easily detectable [29].
3. Some algorithms will only output Black & White Images after data is hidden [1].
4. There is limited capacity as it hides in specific bits of an Image (LSB, HSB and specific colours) [31].

One example of an algorithm that does not always show distortion depending on how much data is embedded is the LSB, however as the name suggests, the content is hidden on the least significant bits of our carrier file and applying a mathematical algorithm to identify the LSB capability, the hidden data is [27].

The constraints of every specific algorithm go into consideration to measure the capacity, on the same carrier file different algorithms apply, and each algorithm will undertake a different capacity measure, e.g. using LSB, the least significant bits targeted by steganalysis [34]. The Discrete cosine transform method algorithms in the Steganography Transform Domain are widely brought to investigation as there are weaknesses to be improved and those are its capacity, its distortion and its detectability, depending on how much data is hidden on the carrier file (the capacity) high detectability and distortion can be experienced. Coefficient bits on the transform domain of the file are used to hide content on the stego-image when applying the DCT method [33]. An 8 x 8 block per RGB (Red, Green, and Blue) is an example that converts into 64 coefficients from which the DCT compression and quantisation applies plus LSB to image coefficients substituting them with the secret content. There is a need for more capacity with minimal distortion and detectability so more data can be hidden and secure with good quality stego-image that can be measured by dB.

An example of newly undertaken research on DCT capacity is “*High-capacity steganography: a global-adaptive-region discrete cosine transform approach*” [24] following a region-based technique of embedding based on colour, texture or shape of carrier image. This recently published paper carries out similar research proposals: Improve capacity with minimal impact on detectability. In 2016, a group of researchers published a paper [19] on how to maximise the capacity under the undetectable model of DCT, taking into consideration various aspects of the carrier image and Quantization Index modulation as opposed to Spread Spectrum. Another paper from Nidhi Antony [2] also proposes improvements on capacity by using watermarking and transparent collage as per the Big Data era where the use and need more capacity.

### **2.3.2 High Capacity Applications using DCT**

Chia-Chen Lin & Pei-Feng Shiu [7] achieved a quality above 30db when hiding on the capacity of 90,000 bits for their research to secure data transmitted on the web with DCT, mainly media files and text documents. DCT has functional importance in the compression of media files [15].

### **2.3.3 Measuring Detectability of DCT**

Measuring Detectability starts by analysing the quality of the stego-image (hence distortion) for high-distortion stego-files, the presence of Steganography is assumed. Several mathematical notations illustrate the quality/distortion of a carrier file and the final quality will indicate the first sign of detectability [36].

# **3 Results**

Detectability is one of the main issues in Steganography algorithms, our initial experiment was to highlight how a small hidden message can alter the size of an image, and DCT is applied (Compression method) to diminish the scale between the size of the stego-image and size of Original Image, making this vulnerability less noticeable in different file formats. Table 1 illustrates the results of image Steganography using the algorithm where the stego-message has only 4.096 bytes. The closer the output file is to the original file, the better the security will be as it is less detectable analysing the image size in **Table 1**:

**Table 1.** Comparing BMP to other format experiments

| Image | Original  Size  (bytes) | Stego-Size  (bytes) | Distortion | File Increase | Detectability |
| --- | --- | --- | --- | --- | --- |
| samplePNG1.png | 860.160 | 864,256 | low | 0.47619047619047666% | Low |
| sampleGIF1.gif | 114,688 | 167.936 | low | 46.42857142857143% | Medium |
| sampleBMP1.bmp | 1.163,264 | 1.056,768 | low | -9.54929577464777% | Low |
| sampleJPEG1.jpg | 262,144 | 1.179.648 | low | 349.99999999999994% | High |

Again, the reason behind BMP being the best solution to Steganography on Image files is the fact that the format is pure, it uses bitmap where all bits are present in the file as opposed to other file formats that use techniques of compression by the standard [37]. BMP is also lossless, where the final stego-image pixel values modified are only the ones gathered to embed the data. For this reason, the final file format of an original BMP image changes after embedding the secret information; there could be a smaller file size and not a more prominent image as a result of the extra information added, which generally implies that Steganography applies.

Considering detectability as the main vulnerability on LSB, not on the increased File Size, but on the “Least Significant Bits embedment” that triggers less security when using Statistical Analyses, let us observe what happens with an image characterised by the binary bits below with the following 8 bits 10101010, where line one is the original image, while line 2 shows the stego-image.

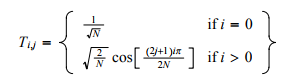
10001101 10011111 11110001 10001111

10000001 10000000 11111110 10101011

1000110**1** 1001111**0** 1111000**1** 1000111**0**

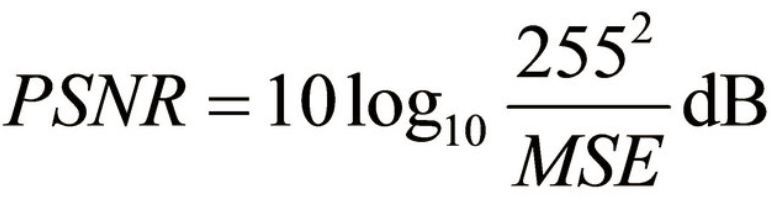
1000000**1** 1000000**0** 1111111**1** 1010101**0**

**Equation 1.** DCT Matrix Representation



The DCT basic algorithm still includes LSB, but it uses matrix complexity and quantisation represented by **Equation 1**. Apart from putting the file formats into detection testing, the frequency power/agility was tested on the file formats to measure security in PSNR as per **Equation 2**, which outputs dB. Results of experiments made with different file formats are in **Table 2**. The higher the PSNR, the higher the security of the algorithm and the larger the image, the higher the capacity.

**Equation 2.** Measuring Security on Image Steganography Formula



**Table 2.** Power Measurement per File Format

| Image | Power in dB | | Security | Capacity |
| --- | --- | --- | --- | --- |
| samplePNG1.png | 9.34dB | Low | | Medium |
| sampleGIF1.gif | 10.86dB | Medium | | Low |
| sampleBMP1.bmp | 65dB | High | | High |
| sampleJPEG1.jpg | 8.99dB | Low | | Low |

The difference between the smaller file and the larger one is 1048.576 bytes

File 1 is 1.352380952380952 smaller than the larger file.

File 2 is 10.14285714285714 smaller than the larger file.

File 3 is our larger file.

File 4 is 4.4375 smaller than the larger file.

Therefore, the bigger the image, the higher the security as there is more space to hide the data and BMP has one of the best performances among other file types including for capacity and security, achieving better results than Nidhi [2] in 2015 and Chia-Chen-Lin in 2010 [7]. As per our experiments, BMP is chosen as the best file to use as it diminished the file size after Steganographyy was applied, and JPG is the one that performed least to use, as using the same algorithm and same stego-message, the file size has increased approximately 350% after Steganography. To demonstrate the proposed approach with a plain 2028 x 1520 pixels jpg image example called Canary (C:\Users\istteffanny.araujo\Desktop\canary.JPG) and a sample message “I am at university today to provide an update of my project”, a 2 x 2 matrix. Two encryption algorithms are in place as an option before transforming the result in binary to distribute later the stegano files generated by the application. RC4 is quicker, and occupies less space as there is a smaller encryption key RSA is slower because the encryption key is secure. The result of the distributed encryption is the four Stegano files with secret data.

From here, there are a few proposed approaches, one is to reassemble the image that has been “cut” and add the secret data into a single image for storage. Another is to embed these fragments into the original raw image adding an even extra layer of security lastly keep the stegano images hidden and not generate the stegano images until needed, and only generate with a particular application or software. Now, capacity enhancement is the main demonstration taking into consideration the sizes of the manifest image versus the sizes of the partitioned images before applying the secret data.

The Original file is a 2028 x 1520 JPG file converted to four 1014 x 760 BMP files to increase the embedding capacity. Considering the four available images to embed data, the total number of pixels has increased to 4056 x 3040, representing a 100% x 100% increase in this example. Comparing **Fig. b** and **Fig. c**, without the hidden message and with respectively, the everyday partitioned images to their final stegano images also pass the raw sight test where no precise detection of the secret message is appointed. The secret message is not detectable in plain sight, and the final stego-image is apparent with the secret data added. **Fig. b** is a partitioned file without any data added and **Fig. c** shows the result of embedding part of the confidential data.



**Fig. b** Plain Image **Fig. c** Image with Secrete data

Finally, **Fig c**. **Fig. d**. **Fig. e** and **Table 3** highlight how the proposed methodology tackles all previously appointed weaknesses with the successful use of BMP that is proven the most efficient together with our distributed DSoBMP-I Steganography prototype, so-called the *New Adaptative Distribution*.

**Fig. d** Efficiency of BMP over other file formats

**Fig. e** Low Detectability of BMP over other formats per file increment after Steganography

**Table 3.** Tackling Limitations Table

| Limitation | Improvement Highlights |
| --- | --- |
| Capacity | Including distributed Steganographyy and the original BMP file type, there are a higher number of images to play, there are more available bits to embed the secret data, and the uncompressed BMP file type has uncompressed bits. They provide better stenographic results than experiments provided in the past. |
| Detectability | The several levels of Steganography using distributed Steganographyy, the combination of encryption of the secret message, and the fact that no fixed partition breaks the original image into an array of newly added images to improve the security and make the secret message less detectable. The “hacker” will not know how many partitions are used to break down the image in each case. Even if a fragment of the partitioned image is found on the original image, the “hacker” would not know a secret encrypted message in those portioned images. If they decided to analyse the partitioned image and found the secret message, it would only be a fragment of it and would be encrypted, which makes it very secure and less detectable. |
| Distortion | BMP has proven the best results along Steganographyy because all bits are present. In contrast, other formats have compression applied and distributed Steganographyy also has improved the look of the final stego-image capability of having distortion, as instead of the full message stored on a single image, it distributes over the set of generated images for attaching the secrete message. |

# **4 Conclusion**

The research brings an algorithm to improve the overall Steganography performance applied in security for several years. Identified the main weaknesses of current techniques, not always equipped to deal with Big Data, it suggested that the DSoBMP-I approach can tackle the low capacity, high detectability and distortion issues within most steganographic algorithms used today, especially DCT. After experimenting with different ideas and image formats, the best image type to embed the data is BMP. This approach has proven to increase capacity by 100% compared to other algorithms where file format is not essential and distributed. All, use a simple 2x2 matrix and this only increases as matrixes increase and has proven to output more power than past papers investigated like Nidhi’s research appointed in the text. The detectability and distortion are also not visible in plain sight, and therefore, it recommends that the next approach for our needs to secure Copywrite materials and Big Data can benefit from the DSoBMP-I methodology proposed here.

**Acknowledgement** This research is from the Intelligent System Research Centre at London Metropolitan University. The authors would like to thank Dr Qicheng Yu from the School of Computing and Digital Media at London Metropolitan University (United Kingdom) for their helpful comments and observations.

**Publisher’s Note** Springer Nature remains neutral concerning jurisdictional claims in published maps

and institutional affiliations.

# **References**

[1] Al-Turfi, M. (2012). Text Realisation Image Steganography. *International Journal of Engineering*. 6 (1), p1-9.

[2] Antony, N. (2015). Improved Capacity College Steganography Using Discrete Cosine Transformation. International Journal of Scientific & Engineering Research. 6 (11), p1060-1064.

[3] Bisk. (2017). *The History of Information Security.* Available: https://www.villanovau.com/resources/iss/history-of-information-security/#.WPSEv\_nyuM9.

[4] Borse, D. And Patil, S. (2015). Review on Transform Domain Steganographic Techniques (DCT & DWT). *International Journal of Innovative Research in Computer and Communication Engineering*. 3 (12), p12466-12473.

[5] Chandramoulia, R. and Memon, N. (2015). Steganography Capacity: A Steganalysis Perspective. a Department of E.C.E., Stevens Institute of Technology. 1 (1), p1-5.

[6] Chen, J. et al. (2013). A Space Increased Reversible Information Hiding Technique by Reducing Redundant Recording. *Procedia Computer Science*. 17 (1), p229-236.

[7] Chia-Chen, L. And Pei-Feng, S. (2010). DCT-based Reversible Data Hiding Scheme. *Journal of Software*. 5 (2), p214-224

[8] Conway, Maura. (2004). Code Wars: Steganography, Signals, Intelligence, and Terrorism. *Technology and Terrorism*. 16 (2), p171-191.

[9] Dave, H. P. (2012). Steganography Technique Based on DCT Coefficients. International Journal of Engineering Research and Applications, 713-717.

[10] Elham Ghasemi, J. S. (2011). Capacity Image Steganography using Wavelet Transform and Genetic Algorithm. Proceeding of the International Multiconference of Engineers and Computer Scientists, 495-498.

[11] Jahankhani, H. et all. (2008). Conference proceedings. In: Jahankhani, H. et all *Global E-Security*. London: Springer. p23-25.

[12] Jason Hiney, T. D. (2015). Using Facebook for Image Steganography. George Mason University, 2.

[13] Kaitezenbessier, S. And Petitcolas, F (2016). Information Hiding. US: Artech House. p40-41

[14] Kalaivanan.S, A. A. (2015). A Survey on Digital Image Steganography. International Journal of Emerging Trends & Technology in Computer Science, 30-33.

[15] Kessler, G. And Hosmer, C. (2011). *Chapter 2 – An Overview of Steganography.* Available: http://www.sciencedirect.com/science/article/pii/B9780123855107000023.

[16] Kim, C. & Yang, CN. Multimed Tools Appl (2015) 74: 5189. <https://doi.org/10.1007/s11042-013-1667-6>

[17] Korus, P., Białas, J. & Dziech, A. Multimed Tools Appl (2014) 68: 59. <https://doi.org/10.1007/s11042-011-0986-8>

[18] Leiner, B. M. et al. (2017). *Brief History of the Internet.* Available: [http://www.internetsociety.org/internet/what-internet/history-internet/brief-history-internet.](http://www.internetsociety.org/internet/what-internet/history-internet/brief-history-internet.%20)

[19] Mao, J. et al. (2016). A method to estimate the steganographic capacity in the DCT domain based on the MCUU model. Wuhan University Journal of Natural Sciences. 21 (4), p283–290.

[20] Manda, N. G. (2010). Image Authentication Technique in Frequency Domain based on Discrete Fourier. Proceedings of ICCS, 144-147.

[21] Mazurczyk, W., Karaś, M., Szczypiorski, K. et al. Multimed Tools Appl (2016) 75: 13521.

[22] Mazurczyk, W., Szaga, P. & Szczypiorski, K. Multimed Tools Appl (2014) 70: 2139. <https://doi.org/10.1007/s11042-012-1224-8>

[23] Neeta Nain, J. S. (2012). Steganography using Odd-even Based Embedding and Compensation Procedure to Restore Histogram. Proceedings of the World Congress on Engineering and Computer Science, 24-26.

[24] Rabie, T. And Kamel, I. (2017). High-capacity Steganographyy: a global-adaptive-region discrete cosine transform approach. Multimedia Tools and Applications. 76 (5), p6473–6493.

[25] Ramapriya, B. (2017). An Improved Approach of Text Steganography in Application with Rotational Symmetry. *International Journal of Innovative Research in Computer and Communication Engineering*. 5 (7), p12939-12947.

[26] Rana, S. & Sur, A. Multimed Tools Appl (2018). <https://doi.org/10.1007/s11042-018-7024-z>

[27] Sharma, A. And Jain, N. (2013). Steganography & Cryptography for Regimented Data Hiding System: A Review. *International Journal of Science and Research*. 4 (4), p3119-3124.

[28] Super, A et All. (2005). The Rise of Steganography. *Proceedings of Student/Faculty Research Day*. 1 (1), 1.

[29] Shun Zhang, T.-g. G.-s. (2014). A High Capacity Reversible Data Hiding Scheme Based on Multi-Level Integer DWT and Histogram Modification. International Journal of Digital Crime and Forensics, 51-64.

<https://doi.org/10.1007/s11042-015-2740-0>

[30] Scheiner, S. (2015). Why is BMP used in Steganography? Available: https://www.quora.com/Why-is-bmp-used-in-steganography. Last accessed 12th Jan 2018.

[31] Tataru, R. et al. (2015). Is Hidden Data Safe? Analysis of the Public Crypt&Hide-Steganos Application. *Proceedings of the Romanian Academy*. 16 (1), p299-312.

[32] Thampi, S. M. (2007). Information Hiding Techniques: A Tutorial Review. LBS College of Engineering, 1-15.

[33] Tsai, C.-W. L.-H. (2010). A Lossless Data Hiding Method by Histogram Shifting Based on an Adaptive Block Division Scheme. Department of Computer Science, National Chiao Tung University, 1-14.

[34] V. Britanak, P. C. (2006). Discrete Cosine and Sine Transforms General Properties, Fast Algorithms and Integer Approximations. San Diego: Elsevier.

[35] Zielińska, E. et all. (2015). Development Trends in Steganography. *Institute of Telecommunications*. 1 (1), p1-13.

[36] Wayner, P. (2009). Disappearing Cryptography. Information Hiding and Watermarking. 3rd ed. Burlington: Elsevier. p337-353.

[37] Westfeld, A. (2004). F5—A Steganographic Algorithm High Capacity Despite Better Steganalysis. Technische Universit¨at Dresden, 289-302.

**Istteffanny Isloure Araujo** holds a First-Class B.Sc. (2008) in Computer Science, a Distinction M.Sc. in Computer Forensics and IT Security accredited by the British Computing Society (2014) and she is currently working on achieving a PhD in Computing. All from London Metropolitan University, United Kingdom. She has published three papers entitled “Enhancement of Capacity, Detectability and Distortion of BMP, GIF and JPEG Images with Distributed Steganography”, “Protecting against Eavesdropping on Mobile Phones to Snip Data with Information Security Awareness and Steganography Principles” and “[Vulnerability Exploitations Using Steganography in PDF Files](http://ijcna.org/abstract.php?id=442)”. She works as an Associate Lecturer on Security, Networks, Cloud Services and Programming. Her research interests include improving Information Security for Big Data and tackling Digital crime investigations on standard and mobile devices.

 **H. B. Kazemian** received a B.Sc. in Engineering from Oxford Brookes University, UK, an M.Sc. in Control Systems Engineering from the University of East London, UK, and a PhD in Learning Fuzzy Controllers from Queen Mary University of London, UK, in 1985, 1987 and 1998 respectively. He is currently a full professor at London Metropolitan University, UK. He worked for Ravensbourne College University Sector, UK, as a senior lecturer for eight years. Previous lecturing experience includes the University of East London, UK, University of Northampton, UK, and Newham College, UK. Research interests include AI and ML applications to cybersecurity. Prof. Kazemian is a Fellow of the Institution of Engineering and Technology FIET (formerly IEE) the UK, Chartered Engineer (C.Eng.) The UK, and Fellow of the British Computing Society (BCS) the UK.