Steganography in image files: A survey

Nagham Hamid, Abid Yahya, R. Badlishah Ahmad, Dheiaa Najim, Lubna Kanaan

School of Computer and Communication Engineering University Malaysia Perlis (UniMAP)
02000 Kuala Perlis, Perlis, Malaysia
Ministry of Science and Technology Iraq, Baghdad

Abstract: Steganography is one of the methods used for the hidden exchange of information. It is the art and science of invisible communication, which strives to hide the existence of the communicated message into media such as text, audio, image and video without any suspicion. Different steganographic techniques with properties of undetectability, robustness and capacity had been proposed in the past. The focus in this paper is on the use of an image file as a carrier, and hence, the taxonomy of current steganographic techniques for image files has been presented. These techniques are analyzed and discussed in terms of their ability to hide information in image files, the amount of the information that can be hidden, and the robustness to different image processing attacks. Steganalysis, which is the science of attacking steganography, is not the focus of this survey; nevertheless it has been briefly discussed.

Key words: Image files, Spatial Domain, Steganography, Survey, Taxonomy, Transform Domain.

INTRODUCTION

In this modern era, computers and the internet are major communication media that connect different parts of the world as one global virtual world. As a result, people can easily exchange information and distance is no longer a barrier to communication. However, the safety and security of long-distance communication remains an issue. This is particularly important in the case of confidential data. The need to solve this problem has led to the development of steganography schemes. Steganography is a powerful security tool that provides a high level of security, particularly when it is combined with encryption (Cheddad, Condell, Curran, & Kevitt, 2010).

Though steganography differs from cryptography; yet they have many points in common. The goal of cryptography is to secure communications by changing the data into a form that an eavesdropper cannot understand. Steganography techniques, on the other hand, tend to hide the existence of the message itself, which makes it difficult for an observer to figure out where the message is. In some cases, sending encrypted information may draw attention, while invisible information will not. Accordingly, cryptography is not the best solution for secure communication; it is only part of the solution. Both sciences can be used together to better protect information. In this case, even if steganography fails, the message cannot be recovered because a cryptography technique is used as well (EL-Emam, 2007).

Watermarking and fingerprinting, among technologies related to steganography, are basically used for intellectual property protection (Morkel, Eloff, & Olivier, 2005).

A digital watermark is a signal permanently embedded into digital data (audio, images, video, and text) that can be detected or extracted afterwards to confirm the authenticity of the data. The watermark is hidden in the host data in such a way that it cannot be removed without deterioration the host medium. Though this method keeps the data accessible, but it is permanently marked (Lu, 2005). The hidden information in a watermarked object is a signature referring to the origin or true ownership of the data in order to ensure copyright protection. Figure 1 shows that ALJAZEERA TV channels typically have their logos watermark for their broadcasting. In the case of fingerprinting, different and specific marks are embedded in the copies of the work that different customers are supposed to get. In this case, it becomes easy for the intellectual property owner to identify such customers who give themselves the right to violate their licensing agreement when they illegally transmit the property to other groups (Anderson & Petitcolas, 2006).

The performance of a steganographic system can be measured using several properties. The most important property is the statistical undetectability (imperceptibility) of the data, which shows how difficult it is to determine the existence of a hidden message. Other associated measures are the steganographic capacity, which is the maximum information that can safely embedded in a work without having statistically detectable objects (Cox, Miller, Bloom, Fridrich, & Kalker, 2007), and robustness, which refers to how well the steganographic system resists the extraction of hidden data.

Nearly all digital file formats, with a high degree of redundancy, are known for their being used for
Steganography, the redundant parts refer to those parts capable of change without any possibility to detect the alteration. Image and audio files are the best cases in point (Morkel et al., 2005). In fact, digital images are the most used carrier file formats owing to their popularity on the internet. The existing work is revolved around steganography in image files. Other types of steganography are not discussed here, such as video, audio, and linguistic.

There are specific terms that are commonly used by the information hiding committees. Throughout this paper, the term ‘cover image’ is used to describe the selected image to hide the secret data. The image with embedded information is characterized as ‘stego-image’. However, the image processing and statistical analysis efforts to break steganography algorithms are known as ‘steganalysis’ or ‘attacks’.

The present work surveys some related literature about steganography; it gives taxonomy for the current steganographic techniques for image files. These techniques are analyzed and discussed not only in terms of their ability to hide information in image files but also according to how much information can be hidden, and the robustness to different image processing attacks. Finally, the paper presents a brief account about the science that tries to defeat and break different steganography algorithms; steganalysis.

![AL-JAZEERA channel visible watermark.](image1)

**Steganography History:**

Throughout history, people have hidden information in different ways. Actually, steganography is not new; its history is as deep as the history itself. The word ‘steganography’ was basically derived from the Greek words with the meaning “covered writing”. Soon after, researchers used it for thousands of years in various manners (N. Provos & Honeyman, 2003). During the 5th century BCE, the Greek tyrant Histiaeus was taken as a prisoner by King Darius in Susa. Histiaeus needed to send an abstruse message to his son-in-law, Aristagoras, who was in Miletus and in order to do this, Histiaeus shaved a slave’s head and tattooed the message on his scalp. As soon as the slave’s hair grew sufficiently to conceal the tattoo, he was sent to Miletus with the message (Johnson, 1995). In ancient Greece, another method was to peel the wax off a wax-covered tablet, then write a message and to have the application of the wax again. The one in charge to receive the message would simply need to get rid of the wax from the tablet to see the message.

Invisible ink was another popular form of steganography. Ancient Romans had their way in writing between the lines by using invisible ink, and by using substances such as fruit juice, urine, and milk. Using Invisible ink, though seems harmless, a letter might reflect a very different message written between the lines. Invisible ink was used as recently as World War II (Cheddad et al., 2010).

In addition to invisible ink, the Germans used the Microdot technique during the Second World War. Information, particularly photographs, was made so small that they were very difficult to detect (Jamil, 1999).

In 1550, Jerome Cardan, an Italian mathematician, proposed a scheme of secret writing where a paper mask with holes is used. The user of such papers all what he needs is to write his secret message in such holes after placing the mask over a blank sheet of paper. The next step is to remove the mask to fill in the blank parts of the page and in this way the message appears as innocuous text (Jamil, 1999). In the literature (N. F. Johnson & S. Jajodia, 1998; Judge, 2001; N. Provos & Honeyman, 2003) a comprehensive history of steganography can be found.

This technique, steganography, is now highly used in computers files with digital data as the carrier and networks are considered as high-speed dispatch channels. The following section illustrates the taxonomy of steganographic techniques for image files, including an overview of the most important steganographic techniques for digital images.
Steganography Applications:

Steganography can be encountered in a lot of useful applications. The copyright control of materials, the enhancement of the robustness of an image search engines and smart IDs (identity cards) where the details of individuals are embedded in their photographs are all cases in point. Further applications include video–audio synchronization, the safe circulation of secret data of companies, TV broadcasting, TCP/IP packets, in which a unique ID is embedded in an image for the purpose of analyzing the network traffic of particular users (N. F. Johnson & S. Jajodia, 1998). In this respect, some modern applications are demonstrated by (Petitcolas, 2000), one of which is in Medical Imaging Systems. In such systems, a separation is recommended between patients’ image data or DNA sequences and their captions for security or confidentiality reasons, e.g., physician, patient’s name, address and other particulars. Hence, embedding the patient’s information in the image could be a useful safety measure and helps in solving such problems.

Accordingly, steganography helps to provide an ultimate guarantee of authentication that no other security tool can provide. In this manner, an LSB embedding technique is proposed, that embeds electronic patient records via using a bi-polar multiple-base data hiding technique (Shaou-Gang, Chin-Ming, Yuh-Show, & Hui-Mei, 2000). Furthermore, the idea of concealing patients’ data in digital images is introduced and based on the notion that steganography can be part of the normal printing process (Anand & Niranjan, 1998; Yue, Chang-Tsun, & Chia-Hung, 2007). As an example, is the Japanese firm Fujitsu, which developed a technique to encode data into a printed picture that is invisible to the human eye. However, such a picture can only be decoded by a mobile phone with a camera. This technique helps users to use their cellular phones in capturing the encoded data (Cheddad et al., 2010).

Digital technologies have swept the confidence in the integrity of visual imagery (Farid, 2009); a matter that motivated researchers to conduct research on digital document forensics. In 2009, Cheddad and his colleagues propose a steganographic scheme which protects scanned documents from forgery using self-embedding techniques. It also allows legal or forensics experts to access the original document though it is manipulated (Cheddad, Condell, Curran, & Kevitt, 2009).

Image Steganography Techniques:

This section gives an overview of steganographic techniques applicable to specific image formats. An image is defined as an arrangement of numbers and such numbers usually stand for different light intensities in different parts of the image. The numeric description takes the form of a lattice where the individual points given the name ‘pixels’. In a color scheme, the number of bits is known as the bit depth and this basically refers to the number of bits assigned to each pixel (Morkel et al., 2005). The most prominent image formats, exclusively on the internet, are the graphics interchange format (GIF), joint photographic experts group (JPEG) format, and to a lesser degree, the portable network graphics (PNG) format. The important issue to discuss here is that most of the steganographic techniques attempt to exploit the structure of these formats. However, some literary contributions use the bitmap format (BMP) simply because of its simple and uncomplicated data structure (Cheddad et al., 2010).

There are number of approaches in classifying image steganographic techniques. These approaches can be classified in accordance with the type of covers used with secret communications. Another possibility is done via sorting such approaches depending on the type of cover modification already applied in the process of embedding. The second approach is adopted for the taxonomy presented in this paper, although in some cases an exact classification is not possible. In general, the process of embedding can be defined as follows; a graphical representation is given in Figure 2.

Fig. 2: General model for steganographic system.
Let $C$ denotes the cover carrier, and $C^-$ the stego-image. Let $K$ represents an optional key (as a seed used to encrypt the message or to generate a pseudo-random noise, which can be set to $\emptyset$ for simplicity), and let $M$ be the message to be sent. Then, $E_M$ and $E_X$ represent an embedded message extracted message respectively. Therefore,

$$E_M: C \oplus M \oplus K \rightarrow C^- \quad (1)$$

$$E_X (E_M(c, m, k) \approx m, \forall c \in C, m \in M, k \in K \quad (2)$$

To distinguish between different steganographic techniques in a wide sense, one must take into consideration both the methods that modify the image and those that modify the image file format. However, the modifications to the file format are less robust (Kruus et al., 2003). The important issue to mention here is the main role compression usually plays when it comes to deciding which steganographic algorithm is better. Though lossy compression methods result in smaller image file sizes, they increase the possibility of the partial loss of an embedded message because surplus image data is to be eliminated in these techniques. Lossless compression does not compress the image file as much (Prasad, Janeyulu, Krishna, & Nagaraju, 2009). As a result, researchers have come up with different steganographic algorithms that suit such compression types. Steganographic techniques that modify image files for hiding information include the followings:

- Spatial domain;
- Transform domain;
- Spread spectrum;
- Statistical methods; and
- Distortion techniques.

Steganographic techniques that modify the image file format involve file embedding and palette embedding. In addition, there are techniques that modify the elements in the visual image including:

- The image generation technique; and the image element modification technique.

Finally, a special type of spatial and transform domain techniques, the adaptive steganography, represents the main concern of the current research. The following sections illustrate the taxonomy of steganographic techniques for image files, including an overview of the most important steganographic techniques for digital images.

A. Steganography in the image spatial domain:

Spatial domain steganographic techniques, also known as substitution techniques, are a group of relatively simple techniques that create a covert channel in the parts of the cover image in which changes are likely to be a bit limited when compared to the human visual system (HVS). One of the ways to do so is to hide information in the least significant bit (LSB) of the image data (Kruus et al., 2003).

This embedding method is mainly based on the fact that the least significant bits in an image can be thought of as random noise, and consequently they become not responsive to any changes on the image (Chandramouli, Kharrazi, & Memon, 2004).

When hiding the message bits in the image using LSB algorithms, there are two schemes, namely sequential and scattered. The LSBs of the image, in the sequential embedding scheme are replaced by the message bits, whereas in the case of the scattered embedding scheme, the message bits are randomly scattered throughout the image using a random sequence to control the embedding sequence (Juneja & Sandhu, 2009).

The well-known steganographic tools based on LSB embedding are different as far as the way they hide information is concerned. Some of them change the LSB of pixels randomly, others modify pixels not in the whole image but in selected areas of it, and still others increase or decrease the pixel value of the LSB, rather than change the value (Cheddad, 2009). A lot of steganographic tools using the LSB based steganographic technique, such as Steghide, S-tools, Steganos, etc, are available on the Internet (Johnson, 2009). These techniques can achieve a high capacity; however, they do not provide robustness against even simple alteration on stego images and are easily detected.

Several variations on the basic LSB techniques have been described by (Johnson & Katzenbeisser, 2000). They also describe a substitution technique for embedding a secret message into the LSB bits of the palette of GIF or BMP image format using steganography. However, their work has not included test images that can allow readers to visualize the concepts. Their survey concludes the evaluation without recommendations or enhancements.

In 2003, a group of researchers maintain that despite the fact that flipping the LSB of one pixel in a JPEG image incurs a slight change (Fridrich, Goljan, & Hogea, 2003); however, changes can still be detected. While
their experiments on the DCT coefficients prove promising results and forward researchers’ concentration towards this type of image. Actually acting at the level of DCT makes steganography less apt to statistical attacks and more robust.

A novel steganographic technique is presented by (D.-C. Wu & Tsai, 2003), in which the difference value between two neighboring pixels are used to decide how many secret bits should be embedded. In their scheme, a cover image is partitioned into non-overlapping blocks of two consecutive pixels. Then in each block, the difference value is calculated from the two pixels values. All possible difference values are grouped into a number of ranges. The selection of the range intervals is based on the characteristics of human vision's sensitivity to gray value variations from smoothness to contrast. The difference value then is replaced by a new value to embed one value of the secret message. This method provides an easy way to produce a more imperceptible result than those given by simple least-significant-bit replacement methods. It is worth mentioning that the pixel-value differencing (PVD) steganography can embed a large amount of secret bits into images with high imperceptibility, since it makes use of the characteristics of human vision sensitivity. On the other hand, remarkable steps in the histogram of pixel differences reveal the existence of a secret message.

To provide larger hiding capacity and to reduce the stego-image distortion, a side match approach is proposed to embed secret data, where the number of bits to be embedded in a pixel is determined by the difference between the pixel and its upper and left side pixels. To estimate the degree of smoothness or contrast of pixels, the proposed method exploits the correlation between neighboring pixels. Such that, if the pixel is located in edge area, then it may tolerate larger changes than those in smooth areas (Chang & Tseng, 2004). In their approach, since the number of bits to be embedded in a pixel is decided by the difference between the pixel and its upper and left side pixels, this will limit the hiding capacity depending on the image texture being used as a cover image. In other words, their scheme is not applicable for all images; there must be a prior study for the cover image that gives the best results.

In 2005, a steganographic scheme is proposed; by combining pixel-value differencing and LSB substitution to embed secret data into still images (H. C. Wu, Wu, Tsai, & Hwang, 2005). Their scheme embeds more secret data into edged areas than smooth areas in the cover image and has a better image quality by using pixel-value differencing (PVD) method alone. To increase the capacity, the secret data is hidden in the smooth areas by using an LSB method with the edged areas still using the PVD method. The experimental results reveal that the proposed method results in stego-images with an acceptable quality and provide a large embedded secret data capacity (H. C. Wu et al., 2005). Anyhow there are still shortcomings in their approach. First, in the embedding process, their (PVD+LSB) approach is too conformable to the LSB approach. Second, both LSB and (PVD+LSB) approaches are easily detected by (Fridrich, Goljan, & Rui, 2001).

In 2008, a new steganographic method is suggested by (Wang, Wu, Tsai, & Hwang, 2008). The method exploits the remainder of two successive pixels to record the information of secret data with more flexibility, to be able to derive the optimal remainder of the two pixels at the least possible distortion. By implementing this scheme, the embedding effect which appears in the stego-image can be significantly reduced in comparison with the scheme presented by (D.-C. Wu & Tsai, 2003). Experimental results obtained by (Wang et al., 2008), show that the proposed scheme has a much better performance in terms of stego-image quality than that proposed by (D.-C. Wu & Tsai, 2003). However, some of the researchers seem not to take into consideration the features of edge embedding enough (Wang et al., 2008; D.-C. Wu & Tsai, 2003; H. C. Wu et al., 2005). At the same time, there are some methods proposed mainly to overcome this drawback, but unfortunately they result in propagated error and lower embedding capacity (Chang & Tseng, 2004; Park, Kang, Shin, & Kwon, 2005).

To provide better stego-image quality and larger embedding capacity, a novel steganographic method is proposed to improve the multi-pixel differencing based on modeled LSB substitution (Liao, Wen, & Zhang, 2011). Similar to the schemes proposed by (Ki-Hyun, Kyeyoung-Ju, & Kye-Young, 2008; Yang & Wang, 2006), a four-pixel block with three difference values is implemented. The average value of three difference values is exploited to distinguish between edge and smooth areas, and to estimate how many secret bits could be embedded into the block. Subsequently, secret bits are embedded into each pixel in the block by adopting modified LSB substitution method. The proposed method considers the features of edge precisely, such that the pixels in edge areas can tolerate much more changes without making perceptible distortion (Liao et al., 2011). Their proposed method achieves an acceptable capacity and imperceptibility. There is still a trade-off between embedding capacity/quality and attack-resistance, and it suffers attack resistance for obtaining higher embedding capacity/quality.

In what has been stated so far, one can conclude that the resulting changes to the cover image using LSB techniques are very difficult to be recognized by the human eye due to being too slight. Moreover, such techniques are simple and popular. The drawback of this technique is that it uses each pixel in the image. As a result, if lossy compression is used, some of the hidden information might be lost. The invention of the LSB embedding is a big accomplishment; its minimal resistance to the attacks forced researchers to invest using it in other applications, such as that of the frequency domain (Cheddad et al., 2010; Geetha, Kablian, Chockalingam, 2013).
& Kamaraj, 2011; Noda, Niimi, & Kawaguchi, 2006). The next section explains data embedding in the image transform domain.

**B. Steganography in the image frequency domain:**

Transform domain embedding can be defined as a domain of embedding for which a number of algorithms have been suggested. The process of embedding data in the frequency domain of a signal is much stronger than embedding principles that operate in the time domain. It is worth saying that most of the strong steganographic systems today operate within the transform domain (Johnson & Katzenbeisser, 2000).

These techniques have an advantage over LSB techniques because they hide information in areas of the image that are less exposed to compression, cropping, and image processing. Some transform domain techniques do not seem dependent on the image format and they may outrun lossless and lossy format conversions (Kruus et al., 2003). Stego methods in transform domain hide data in the coefficients of the represented domain, such as discrete Fourier transforms (DFT), discrete cosine transforms (DCT), and discrete wavelet transforms (DWT) (N. F. Johnson & S. Jajodia, 1998; Johnson & Katzenbeisser, 2000).

In the 2-dimensional DCT phase, each 8×8 non-overlapping block is transformed into the DCT domain by using the 2-dimensional DCT (kekre, Mishra, Shah, Shah, & Thakkar, 2012).

\[
F(u,v) = \frac{c(u)c(v)}{4} \sum_{x=0}^{7} \sum_{y=0}^{7} \cos\left(\frac{(2i+1)ux}{16}\right) \cos\left(\frac{(2j+1)vy}{16}\right) f(i,j)
\]

Where 
\[
c(e) = \begin{cases} 
\frac{1}{\sqrt{2}}, & \text{if } e = 0 \\
0, & \text{if } e \neq 0
\end{cases}
\]

\(F(u,v)\) and \(f(i,j)\) stand for a DCT coefficient at the \((u,v)\) coordinate and a pixel value at the \((i,j)\) coordinate, respectively. \(F(0,0)\) is the DC component, which corresponds to an average intensity value of each block in the spatial domain. \(F(u,v)\) is the AC component, in which \(u \neq 0\) and \(v \neq 0\). For data reduction during the quantization phase, DCT coefficients are quantized by using the standard quantization table, as shown in Figure 3, this table shows that the upper left values in the quantization table are too small to make a large modification. In contrast, the lower right values in the table are large enough to be altered. This merit can be invested to camouflage the human vision system (HVS), since the latter is much more sensitive to the values in low-frequency components than those in the higher frequencies. Thus, distortion in high-frequency components is visually acceptable and imperceptible (C.-C. Lin & Shiu, 2010; Y.-K. Lin, 2012). Although embedding at the DCT level is a very successful and powerful tool, but if coefficients are not carefully selected, some artifacts due to data embedding will be noticeable.

For JPEG images, DCT is used; the pixels can be converted with such mathematical processing as described by Eq. (3). The quantization phase of the compression is counted as the next step. It is further considered a biological property where the human eye is imposed. Basically, the human eye is known for being capable of identifying small differences in brightness over a relatively large area. The same does not apply when considering the distinction between different strengths in high-frequency brightness (Morkel et al., 2005). Consequently, the strength of higher frequencies can be reduced without any change in the image appearance. The JPEG format is done by dividing all the values in a block via a quantization coefficient, so the results are made approximate to integer values.

The last point is to encode the coefficients by using Huffman coding just to reduce the size. Previously, it was believed that steganography could not be used with JPEG images owing to the lossy compression, which results in parts of the image data being altered. JPEG images are the products of digital cameras, scanners, and other photographic image capture devices. This is simply why concealing secret information in JPEG images might provide a better disguise. Data in most of the steganographic systems seems to be embedded into the non-zero discrete cosine transform (DCT) coefficients of JPEG images; steganography based on DCT is illustrated in Figure 4.

| 16 | 11 | 10 | 16 | 24 | 40 | 51 | 61 |
| 12 | 12 | 14 | 19 | 26 | 38 | 60 | 55 |
| 14 | 13 | 16 | 24 | 40 | 57 | 69 | 56 |
| 14 | 17 | 22 | 29 | 51 | 87 | 80 | 62 |
| 18 | 22 | 37 | 56 | 68 | 109 | 103 | 77 |
| 24 | 35 | 55 | 64 | 81 | 104 | 113 | 92 |
| 49 | 64 | 78 | 87 | 103 | 121 | 120 | 101 |
| 72 | 92 | 95 | 98 | 112 | 100 | 103 | 99 |

Fig. 3: Standard quantization table.
Fig. 4: Steganography- Based on DCT.

The major JPEG steganographic methods can be described as follows:

**JSteg:** JSteg is a classic JPEG steganographic tool that embeds secret messages in the transform domain using the DCT (Johnson & Katzenbeisser, 2000). It functions to hide the secret data in a cover image by simply exchanging the LSBs of non-zero quantized DCT coefficients with secret message bits. The software embeds the secret message into the lowest-order bits of all nonzero frequency coefficients that result after the DCT has been applied. The results are then Huffman encoded to produce the stego-image in JPEG format (Kruus et al., 2003). Although the algorithm stood strongly against visual attacks, it was found that examining the statistical distribution of the DCT coefficients shows the existence of hidden data (Cheddad et al., 2010; N. Provos & Honeyman, 2003). JSteg is easily detected using the $\chi^2$-test. Moreover, since the DCT coefficients need to be treated with sensitive care and intelligence the JSteg algorithm leaves a significant statistical signature (Cheddad et al., 2010).

**F5:** The F5 steganographic algorithm was introduced by Westfeld. This algorithm embeds data into the non-zero AC DCT coefficients by reducing their absolute values by 1 rather than by replacing the LSBs of quantized DCT coefficients with the message bits (Westfeld, 2001). Due to the authors’ argument, the use of the chi-square ($\chi^2$ attack) can never detect this type of embedding (Westfeld & Pfitzmann, 2000). Later, a steganalysis method is proposed by (Fridrich & Goljan, 2002), which exploits the natural distribution of DCT coefficients. The proposed method does detect F5 contents, disrupting F5’s survival.

**OutGuess:** OutGuess is provided by Provos as a UNIX source code; it has two widely known released versions (Niels Provos, 2001). The first one is the OutGuess-0.13b, which is exposed to statistical analysis. The second is OutGuess-0.2, which includes the ability to safeguard statistical properties. Hereafter, OutGuess refers to OutGuess-0.2. There are two stages representing the embedding process of OutGuess. The first stage involves secret message bits embedding along a random walk into the LSBs of the quantized DCT coefficients while skipping 0s and 1s. The second stage entails some modifications made to the coefficients already left during the process of embedding. This helps the global DCT histogram of the stego image match that of the cover image. OutGuess cannot be subjected to $\chi^2$-attack. The inventor of OutGuess proposed a counter attack against this algorithm (N. Provos & Honeyman, 2003) by applying an extended version of the $\chi^2$-test to select Pseudo randomly embedded information in JPEG images.

**MB:** Model-based steganography (MB) can be defined as a general framework for conducting both steganography and steganalysis by simply using a statistical model of the cover media (Sallee, 2004). The MB method for JPEG images is capable of having high message capacity while remaining secure against many first-order statistical attacks (C.-C. Lin & Shiu, 2010).

**YASS:** Yet another steganographic scheme (hence YASS) belongs to JPEG steganography; however, it does not conceal data in JPEG DCT coefficients directly (Solanki, Sarkar, & Manjunath, 2007). Instead, an input image in the spatial domain is divided into blocks with a fixed large size, called big blocks (or B-blocks). Within each B-block, an $8 \times 8$ sub-block, known as embedding host block (or H-block) will be randomly selected. Then, via using error correction codes, secret data is encoded and embedded in the DCT coefficients of the H-blocks. Finally, the entire image is compressed and distributed as a JPEG image after inversing DCT on the H-blocks (B. Li, He, Huang, & Shi, 2011).

As far as the DWT is concerned, steganography techniques in transform domain intend to hide data by modifying the DWT coefficients. Wavelets are used in the image steganographic model because the wavelet transform clearly partitions the high-frequency and low-frequency information on a pixel by pixel basis. The DWT method is favored over the DCT method, owing to the resolution that the DWT provides to the image at various levels (Reddy & Raja, 2009).

In 2000, a group of writers discuss steganography technique that inserts a secret message into a base layer transmission of a zero-tree based wavelet coder (Yasser Fouda Syed, 1999; Y. F. Syed & Rao, 1999). They intend to hide the message in sign/bit values of irrelevant children of the detail sub bands in nonsmooth regions...
of the image. The Homogenous Connected-Region Interested Ordered (HC-RIOT) coder is used to determine what regions of the image that the message can be embedded in. The coder also decides which wavelet coefficients in the detail sub bands of these regions could be modified. The main feature of this approach is the ability to send steganographic messages in lossy environments with robustness against detection or attack (Areepongsa, Kaewkamnerd, Syed, & Rao, 2000; Areepongsa, Syed, Kaewkamnerd, & Rao, 2000).

In 2006, (Paulson, 2006) reports that a group of scientists at Iowa State University develop an advanced application called Artificial Neural Network Technology for steganography (ANNTS), with the aim of detecting all current steganography methods in the image transform domain, including DCT, DWT, and DFT. (ANNTS) exploits neural networks to statistically scan digital files and then check them for changes in pixel values to be able to detect if there is steganographic hidden information. For example, it is observed that, in an image with no steganography, the number of pixels with even values, based on their position on a gray or color scale, and the number with odd values will be very different. In an image with steganographic hidden data, the numbers are almost equal. This provides a simple statistical test for steganography. Their conclusion comes with the fact that since the (DFT) includes a rounding off error, due to the multiplications performed in the DFT with fixed point arithmetic; this makes DFT inappropriate for steganography applications.

A data hiding technique in the DWT domain is successfully implemented by (Abdelwahab & Hassaan, 2008). In this scheme, both secret and cover images are decomposed using 1st level DWT, each of which is divided into disjoint 4x4 blocks. Then a comparison is made between the blocks of the secret image and the cover blocks to determine the best match. Later, error blocks are produced and embedded into the coefficients of the best matched blocks in the horizontal sub-band of the cover image. Two keys must be communicated: one to hold the indices to the matched blocks in the cover approximation sub-band, and another for the matched blocks in the horizontal sub-band of the cover. The proposed scheme shows high robustness against a lot of image processing operations such as lossy compression, blurring, cropping, median filter, sharpen, and addition of noise. The main drawback of this system is the extracted payload is not totally identical to the embedded version. Since the only extracted and embedded bits belong to the secret image approximation, while setting all the data in other sub images to zeros during the reconstruction process.

One year later, a high capacity and high security steganography system (HCSSD) is presented using the discrete wavelet transform (Reddy & Raja, 2009). In (HCSSD) system the cover and payload are normalized and the wavelet coefficient is obtained by applying discrete wavelet transform. The approximation band coefficient of payload and wavelet coefficient of cover image are fused based on strength parameters alpha and beta. The cover and payload are preprocessed to reduce the pixel range to ensure the payload is recovered accurately at the destination. The capacity of the proposed algorithm is increased as the only approximation band of payload is considered. The Entropy, Mean Squared Error (MSE) are improved with acceptable visual quality for the resultant stego-image. However, in their research, there is no discussion about the robustness of the proposed system against attacks and/or statistical analysis.

In 2011, a promising steganography method is implemented via data embedding within skin region of image, which provides an excellent secure location for data hiding (Shejul & Kulkarni, 2011). In their study, biometric steganography is presented using skin region of images in DWT domain for secret data embedding. Secret data is hidden in one of the high frequency sub-band of DWT by tracing skin pixels in that sub-band. By embedding data in specific regions of an image (skin region), security is enhanced with or without cropping and histogram-based attacks are successfully prevented, the proposed approach provides fine image quality. On the other hand, hiding the secret information in specific regions in the image limits the payload capacity, in addition choosing (skin region) of the image for embedding may add another limitation for the test images that have been used as cover images.

For more details and examples about steganography in the (DWT) domain, the reader is directed to the literature (W.-Y. Chen, 2007; Potdar, Han, & Chang, 2005; Shen, Zhang, Feng, Cao, & Huang, 2007).

C. Spread spectrum image steganography techniques:

Spread spectrum (SS) communication technologies have been developed since the 1950s in an attempt to provide a means of low-probability-of-intercept and anti-jamming communication. Spread spectrum technique can be defined as follows (Pickholtz, Schilling, & Milstein, 1982):

“Spread spectrum is a means of transmission in which the signal occupies a bandwidth in excess of the minimum necessary to send the information; the band spread is accomplished by means of a code which is independent of the data, and a synchronized reception with the code at the receiver is used for despreading and subsequent data recovery”.

Although the power of the signal to be transmitted could be large, the signal-to-noise-ratio in every frequency band will be small. Even if parts of the signal are removed in several frequency bands, enough information should be present in the other bands to recover the signal. Hence, SS makes it difficult to detect and/or remove a signal. This situation is very similar to a steganography system; it tries to spread a secret
message over a cover image in order to make it impossible to perceive (Johnson & Katzenbeisser, 2000). Since the power of the embedded signal is much lower than the power of the cover image, the embedded information becomes imperceptible not only to the human eye but also through computer analysis without access to the original image (Marvel, Boncelet, & Retter, 1999).

The Spread Spectrum Image Steganography (SSIS) is a data hiding communication steganographic system which uses digital imagery as a cover signal. SSIS provides the ability to hide a significant quantity of information bits within digital images while avoiding detection by an observer. The message is recovered with low error probability due the use of error control coding (Marvel, Retter, & Boncelet, 1998b). The general additive embedding scheme can be described as follows (Marvel, Retter, & Boncelet, 1998a):

\[ Y_i = X_i + γ W_i \]  

For \( i = 1, 2, \ldots, N \)

Where \( X_i \) is a sequence of the original data from the cover, \( W_i \) is a pseudo-random sequence generated from a pseudo-random number generator (PRNG) initialized by a secret stego key, \( γ \) is an embedding strength parameter (gain factor), and \( Y_i \) is a sequence of possibly altered data.

The core of SSIS is a spread spectrum encoder. These devices work by modulating a narrow band signal over a carrier. The carrier's frequency is continually shifted using a pseudorandom noise generator feeded with a secret key. In this way the spectral energy of the signal is spread over a wide band, thus decreasing its density, usually under the noise level. To extract the embedded message, the receiver must use the same key and noise generator to tune on the right frequencies and demodulate the original signal. A casual observer won't be able even to detect the hidden communication, since it is under the noise level (Marvel et al., 1999). Figure 5 shows a simplified SSIS encoder, while Figure 6 illustrates the decoding process. During the encoding phase, the message is converted to pseudo-noise which is then added with the cover image to produce the steganographic image (clipping and digitizing are suggested to create the digital steganographic image). To retrieve the secret message, first the image is filtered to find the pseudo-noise and then the message is extracted from the pseudo-noise (Marvel et al., 1999).

![Fig. 5: Simplified SSIS encoder.](image)

![Fig. 6: Simplified SSIS decoder.](image)
In 1996, the communication theory concepts are exploited by (Smith & Comiskey, 1996) to characterize information hiding schemes. Accordingly, three schemes, namely direct sequence, frequency hopping and chirp have been described. In direct sequence scheme, the secret signal is spread by a pseudo-noise code with constant rate called chip rate. On the other hand, in frequency hopping scheme the frequency of the carrier signal is altered in a way that it hops rapidly from one frequency to another. The chirp is a hybrid system that combines the direct and the frequency hopping schemes for the purpose of information hiding. Their experiments conclude that the scheme which is called “frequency hopping” is superior perceptually, and in terms of robustness to accidental removal, to the direct sequence schemes. Direct sequence may be less vulnerable to intentional removal, and wins in terms of computational complexity. In general, the robustness of such a technique is still in question. A noise filter similar to that used in the SSIS decoder given in Figure. 6 will remove the message entirely.

To increase the payload capacity, spread spectrum techniques can be combined with transform embedding by using transformation techniques. A technique based on DFT is described by (Alturki & Mersereau, 2001). The technique embeds the secret data in the DFT domain after permuting the image pixels in the spatial domain. The process of permuting the image pixels adds randomness into the cover image, resulting in a significant increase in the transform coefficients used for the transmission of secret information (Alturki & Mersereau, 2001).

In 2005, a blind image steganography, based on a hybrid direct sequence/frequency hopping (DS/FH) technique, is proposed (Widadi, Ainianta, & Chan Choong, 2005). The proposed Blind Spread Spectrum Image Steganography (BSSIS) is able to recover the hidden information from the stego image, without the need for the original cover image. The blind scheme is accomplished through the insertion of quantization technique and channel estimation. The quantization process occurs at the encoder side while channel estimation is executed at the decoder end, this is similar to the main scheme proposed by (Marvel et al., 1999).

The problem of hiding information in a digital host image via spread spectrum embedding in an arbitrary transform domain is considered by (Gkizeli, Pados, & Medley, 2007). It is proposed to use the minimum-eigenvalue eigenvector of the transform domain host data autocorrelation matrix as the embedding signature; to maximize the possible signal to interference plus noise ratio (SINR). In their scheme, the authors establish that, under a (colored) Gaussian assumption on the transform domain host data, the same derived signature minimizes host distortion for any target message recovery error rate and maximizes the Shannon capacity of the covert steganographic link. Anyway, such a system in which random-noise-like secret message signals are added to the host, a limited number of features cannot always differentiate between plain images and their corresponding stego versions. Using a larger number of higher-order statistics features can enhance the sensitivity of the feature detector, but significantly increases computational complexity.

In 2010, a group of researchers propose using a code division multiple access (CDMA) spread spectrum for both the spatial domain and the transform domain for image steganography in MMS (Singh, Khan, Khan, & Singh, 2010). This method is used for the secure data transfer from a computer to a mobile phone or to another mobile phone through data cable, Bluetooth, Infrared or MMS, etc. Implementing this CDMA spread spectrum technique in spatial domain provides a stego-system with high robustness and more resistance to the attacks compared with LSB and DCT techniques when they are used for steganography in MMS. The obtained experiment results reveal that the spread spectrum detection method can provide high robustness to normal signal manipulation, including compression and additive noise. The resultant PSNR by adopting this technique is greater than 50 DB (Singh et al., 2010).

Relying on the SSIS method, one can embed around 0.03 to 0.17 bits per pixel in 256-level gray-scale image. Robustness can be traded for capacity (Kruus et al., 2003).

Spread spectrum techniques satisfy most of the information hiding systems requirements and especially the robustness against statistical attacks, since the hidden information is scattered throughout the image, while not changing the statistical properties. In general spread spectrum techniques can be used for most steganography applications; although it’s highly mathematical and intricate approach may prove too much for some.

D. Statistical methods:

Statistical steganographic methods choose statistics that can be derived from an image and, based upon a threshold, returns a single bit of information. To send multiple bits, an image is broken into sub-images, each corresponding to a single bit of the message (Kruus et al., 2003).

Statistical steganographic techniques make use of the existence of a “1-bit” steganographic scheme, in which one bit of information is embedded in a digital carrier. This process is done by simply modifying the cover image to make a sort of significant change in the statistical characteristics if a “1” is transmitted, otherwise it is left unchanged (Johnson & Katzenbeisser, 2000).

A statistical steganography algorithm is constructed by (Johnson & Katzenbeisser, 2000) out of Pitas’ watermarking system (Pitas, 1996), which is similar to the Patchwork technique proposed by (Bender, Gruhl, Morimoto, & Lu, 1996). In order to construct I(m)-bit stego-system from multiple “1-bit” stego-system, a cover
is divided into $l(m)$ disjoint blocks $B_1, B_2, \ldots, B_{l(m)}$. A secret bit $m_i$, is inserted into the $i$th block by inserting “1” into $B_i$ if $m_i=1$. Otherwise, the block is not changed in the embedding process. The detection of a specific bit is done by means of a test function to distinguish the modified blocks from the unmodified ones, as given by (Johnson & Katzenbeisser, 2000):

$$f(B_i) = \begin{cases} 1 & \text{block } B_i \text{ is modified in the embedding process} \\ 0 & \text{otherwise} \end{cases}$$

Conveniently, secret digital watermarks provide a basis for a statistical function that can be used to encode a message. Secret digital watermarks are designed to be hard to detect and hard to remove but easy to recover given a key. An encoder will watermark those blocks that correspond to a message value of “1” and leave untouched those blocks that correspond to a message value of “0” (Yu-Kuen, Mei-Yi, & Jia-Hong, 2002).

Statistical steganographic methods in their simplest form, for which sub-images are simply sub-rectangles of the original image, are vulnerable to cropping, rotating, and scaling attacks, along with any attacks that work against the watermarking technique. To counter these attacks, the sub-images could be selected based on picture elements, for example, the faces in a crowd, and error correction coding could be utilized within the message. These defenses can make the statistical steganographic method approximately as robust as the underlying watermarking scheme (Krus et al., 2003).

Since the main focus of this technique is based on making a significant change in the statistical characteristics of the cover image if a “1” is transmitted. This in turn makes this system easily detected by steganalysis, which can involve coding a program that examines the stego-image structure and measures its statistical properties e.g., first order statistics (histograms) or second order statistics (correlations between pixels, distance, direction). Thus, statistical steganographic techniques are less favorable than other steganographic techniques for communicating secret information.

E. Distortion Techniques:

Distortion techniques require knowledge of the original cover image during the decoding process where the decoder functions to check for differences between the original cover image and the distorted cover image in order to restore the secret message. The encoder, on the other hand, adds a sequence of changes to the cover image. So, information is described as being stored by signal distortion (Radhakrishnan, Shanmugasundaram, & Memon, 2002; Reddy & Raja, 2009). Using this technique, a stego-object is created by applying a sequence of modifications to the cover image. This sequence of modifications is selected to match the secret message required to transmit (Katzenbeisser, 2000). The message is encoded at pseudo-randomly chosen pixels. If the stego-image is different from the cover image at the given message pixel, then the message bit is a “1”. Otherwise, the message bit is a “0”. The encoder can modify the “1” value pixels in such manner that the statistical properties of the image are not affected, which is different from many LSB methods (Johnson & Katzenbeisser, 2000).

An early approach to hiding information was to do so in text. Most text-based hiding techniques are of the distortion type. For example, the layout of a document or the arrangement of words might show or reflect the presence of information. Considering one of these techniques, can show the adjustment of the positions of lines and words where spaces and “invisible” characters are added to the text, providing a method of sending hidden information (Johnson & Katzenbeisser, 2000).

To construct data-embedding techniques for formatted text, considerable efforts have been done by (Low & Maxemchuk, 1998; Low, Maxemchuk, Brassil, & O’Gorman, 1995; Low, Maxemchuk, & Lapone, 1998). In their study, text-based steganographic schemes are presented, in which the distance between consecutive lines of text or between consecutive words is utilized to transmit secret data. It should be noted, however, that any steganographic system which uses the text format to transmit information can easily be broken by retyping the document.

Distortion techniques can easily be applied to digital images. Using the same approach as in substitution systems, the sender first chooses $l(m)$ different cover-pixels to be used for information transfer. Such a selection can again be done using pseudorandom number generators or pseudorandom permutations. To encode “0” in one pixel, the sender leaves the pixel unchanged; to encode “1”, a random value is added to the pixel’s color.

An image distortion technique, has been introduced by (Sandford II, Bradley, & Handel, 1996), in which data embedding tries to modify the order of appearance of redundant data in the cover rather than to change values themselves; the embedding process therefore maintains a “pair list” (i.e., a list of pairs of samples whose difference is smaller than a specific threshold). The receiver can reverse the embedding process if he/she has access to the pair list. This list has the same role as the key in cryptography.

Unlike many LSB methods, information hiding by implementing distortion techniques doesn’t disturb any statistical properties of the image. On the other hand, the need for sending the cover image limits the benefits of this technique. As in any steganographic technique, the cover image should never be used more than once. If an
attacker tampers with the stego-image by cropping, rotating, or scaling, the receiver can easily detect the modification. In some cases, if the message is encoded with error correcting information, the change can even be reversed and the original message can be fully recovered.

**F. File Embedding Technique:**

Different image file formats are known for having different header file structures. In addition to the data values, such as pixels, palette, and DCT coefficients, secret information can also be hidden in either a header structure or at the end of the file (Yesna, Karen, & Sos, 2007). For example, the comment fields in the header of JPEG images usually contain data hidden by the invisible Secrets and Steganozorus. On the other hand, Camouflage, JpegX, PGE10, and PGE20 add data to the end of a JPEG image (Cheddad, Condell, Curran, & McKevitt, 2008a).

Image storage formats such as TIFF, GIF, PNG, and WMF have a file header that can be exploited to hide arbitrary information. In this case, that arbitrary data may be a secret message. It is possible to append data to many image storage formats without affecting the image. When the image is processed for display, the image user will decode the image size from the file header, and any tracking information attached to the end of the file will be ignored. Using this technique, it is possible to attach a message of any size to a cover image. However, the message could be removed from the cover image by simply resaving the image in the same file format (Kruus et al., 2003).

The limitations of this method are that despite the large payload, it is not that difficult to identify and defeat, it is weak when lossy compression and image filtering are concerned, and the resaving of the image implies complete loss of hidden data (Cheddad, Condell, Curran, & McKevitt, 2008b).

**G. Pallet embedding:**

In a palette-based image, what matters is the fact that only a subset of colors from a particular color space is used to colorize the image. Researchers believe that every palette-based image format consists of two parts. The first part is a palette that assigns N colors as a list of indexed pairs \((i, c_i)\), assigning a color vector \(c_i\) to every index \(i\), and the actual image data, which specifies a palette index for each pixel, rather than the color value itself. The file size gets decreased via this approach when only a limited number of color values are used in the image (Johnson & Katzenbeisser, 2000).

Two of the most popular formats of palette-base images are the graphics interchange format (GIF) and the bitmap format (BMP). However, owing to the availability of advanced compression techniques, their use has diminished (Xuefeng, Then, & Chang-Tsun, 2005).

In some cases, the palette itself can be used to hide secret information. Because the order of the colors in the palette usually does not matter, the ordering of colors can be used to transfer information. In essence, a hidden message can be embedded using the difference between two colors in the palette (i.e., one secret message bit for every two colors in the palette). Color palettes are used to minimize the amount of information images that are used to represent colors (Kruus et al., 2003; Samaratunge, 2007).

Since steganographic message within the bits of the palette and/or the indices is embedded in the palette-based steganography, one must be careful not to exceed the maximum number of colors (Chih-Hsuan, Zhi-Fang, & Wen-Hsiang, 2004).

One of the popular steganographic techniques for hiding information in the palette, are those that exploit the LSB of the palettes color values to hide the secret information. Since the changes in the LSB do not extensively alter the color values and may not be perceptible by the human eye. The software programs like Ezstego adopt this technique (Wayner, 2002). In this technique, the palette is re-ordered in such a way that the neighboring colors in the palette are perceptually similar, as a prior stage before information embedding. Thus, make the technique more robust in the case that the attacker tends to re-order the palette to remove the embedded information.

However, palette embedding techniques are not very robust in comparison with other steganography techniques. On the other hand, these techniques do not survive simple recording attacks, where the attacker records the palette for the purpose of destroying the message without affecting the image.

**H. Image generation technique:**

Many techniques have been proposed that encrypt messages so that they are unreadable or as secret as possible. As an example, the software application Sam's Big Play Maker hides information by converting the secret text message into a larger and a slightly manipulated text format (Ec, 2010). The same principle can be employed in image creation, in which a message is converted to picture elements and then collected into a complete stego-image. This method cannot be broken by rotating or scaling the image, or by lossy compression. Parts of the message may be destroyed or lost because of cropping, but it is still possible to recover other parts of the message by encoding the message with error correcting information (England, 1997).
Generally, this technique uses pseudo-random images, if a malicious third party detects a group of images passing through a network without any reason for them being there (i.e., random images), he or she may suspect that the images contain secret information and block their transmission (Kruus et al., 2003).

I. Image element modification techniques:

Some steganographic techniques do not try to hide information using the actual elements of the image. Instead, they adjust the image elements in completely undetectable ways, for example, by modifying the eye color or hair color of some person in a photograph. These modifications can then be used to carry the hidden information. It is worth mentioning that this information will survive rotations, scaling, and lossy compression. Moreover, Clipping images may remove part of the embedded message, but there is still a possibility to be recovered through the use of error correcting coding (Kruus et al., 2003).

The feasibility of modifying objects within images as a tactic for hiding information has been discussed by (Bender et al., 2000). It is important to notice that when this method is used, the same cover image must not be used more than once, because the elements used will become apparent. This technique can be achieved manually with any photo editing software. With computer vision systems that recognize objects becoming more viable, the process may be able to become automated in the near future (Kruus et al., 2003).

J. Adaptive steganography:

Adaptive steganography is a special case of the spatial and transform techniques. Moreover, it is introduced as “Statistics-aware embedding” (N. Provos & Honeyman, 2003), “Masking” (N. F. Johnson & S. Jajodia, 1998) or “Model-Based” (Sallee, 2004). In adaptive steganography techniques, global statistical characteristics of the image are basically used before any attempt to deal with its LSB/frequency transformed coefficients. These statistics decide what changes could be made (Tzschoppe, Baeuml, Huber, & Kaup, 2003). A random adaptive selection of pixels actually characterizes this method, relying on the cover image and the selection of pixels in a block with a large standard deviation (STD). The latter is intended to avoid areas of uniform color, such as smooth areas. This behavior makes adaptive steganography seek images with existing or deliberately added noise and images that demonstrate color complexity (Cheddad et al., 2010; Cheddad, Condell, Curran, & Mc Kevitt, 2009).

The model-based technique, MB1, depicted by (Sallee, 2004), generates a stego-image based on a given distributed model, a generalized Cauchy distribution is adopted to result in a minimum distorted stego-image. Unfortunately, this steganographic algorithm is broken using the first-order statistics (Böhme & Westfeld, 2004, 2005). Besides, it can also be detected by the difference of blockiness between a Stegoimage and its estimated version (L. Yu, Zhao, Ni, & Zhu, 2008).

An adaptive steganographic scheme for index-based images has been proposed by (Chang, Tsai, & Lin, 2004). The idea behind this method is to partition the color palette for color image data hiding. The colors in palette (code-words) are grouped into sub-clusters according to the relationship among code-words. The size of the sub-cluster is designed to determine the hiding capacity. In comparison with the traditional LSB method, their experimental results show the performance of the proposed scheme. A better stego-image quality is obtained in comparison with the methods proposed by (Fridrich, 1999; Fridrich et al., 2001). Another advantage of their scheme is a higher hiding capacity is provided.

To break the detection of the spatial domain and the frequency domain steganalysis systems, a genetic algorithm (GA) based method is introduced (Shih, 2007; Yi-Ta & Shih, 2006). The stego-image is generated by artificially counterfeiting statistical features with the aid of genetic algorithm. In their study, they mention that the process should be repeated until a predefined condition is satisfied, or a constant number of iterations are reached. The predefined condition is the situation when the desired hidden message could be correctly extracted. The main drawback of genetic based algorithms, which is time complexity, was not discussed by the authors. In addition, it is not stated whether the process of calculating such a condition is done automatically or including a visual perception. The suggested GA-based rounding error correction algorithm, even as interesting, still needs proof of generality.

A content-based image embedding method based on segmenting homogenous grayscale regions using a watershed method coupled with Fuzzy C-Means (FCM) is proposed by (Jun, Hongru, Xiaolu, & Zhi, 2009). In this method, the secret data is encrypted by chaotic map before embedding. Then the cover image is segmented by watershed algorithm and FCM. After that the feature of each region is extracted and the secret data is embedded into the cover image according to the result of feature extraction. Entropy is then calculated for each region. The obtained entropy values dictated the embedding strength where four LSBs of each of the cover’s RGB primaries were used if it exceeded a specific threshold otherwise only two LSBs for each were used (Cheddad, 2009). The weakness point of this method is its sensitivity to intensity changes which affect severely the extraction of the correct secret bits. As a side note, in (Jun et al., 2009), the authors also reported the use of a logistic map to encrypt the secret bit stream which seems vulnerable to a Chosen-plaintext attack, CPA.
In 2009, a novel steganographic technique is proposed by (Cheddad, 2009). A robust steganographic system called “Steganoflage” is established. Steganoflage advocates an object-oriented approach in which skin-tone detected areas in the image are chosen for embedding where possible. The adopted embedding method is the Reflected Binary Gray Code, RBGC, in the wavelet domain. The obtained results are promising and identify the support and consistency of the developed algorithm. A series of interesting applications are shown, such as combating digital forgery, multilayer security for patients’ data storage and transmission and digital reconstruction of lost signals. In their scheme, the main limitation that must be taken into consideration is choosing embedding in skin-tone regions, which affects to some extent the freedom of selecting the cover image. On the other hand, skin-tone embedding limits the hiding capacity. Then, there is no clear discussion about the achieved payload capacity. Finally, the authors decide to encrypt the image itself, which may attract more attention from eavesdroppers in case of intentional or unintentional attacks.

A technique called the “adaptive more surrounding pixels using” (A-MSPU) technique, which improves the imperceptibility problems of multiple base notational systems (MBNS), has been discussed by (Afrakhteh & Ibrahim, 2010). This technique pays attention to the embedding in the edge areas of a cover image while re-expressing the secret bits in multiple base notational systems. The suggested approach uses the same probability parameter to get the secret bits scattered and it also uses surrounding pixels with the maximum number to determine the capacity of every target pixel. Most steganographic techniques use either three or four adjacent pixels of a target pixel. Their proposed technique is able to utilize all eight adjacent neighbors, which improves the imperceptibility value.

Performance Measure:

As a performance measure for image distortion due to embedding, the well-known peak-signal-to noise ratio (PSNR), which is categorized under difference distortion metrics, can be applied to stego images (Cheddad et al., 2010). It is defined as:

$$\text{PSNR} = 10 \log \left( \frac{c_{\text{max}}^2}{\text{MSE}} \right) \quad (6)$$

Where MSE denotes the mean square error, which is given as:

$$\text{MSE} = \frac{1}{MN} \sum_{x=1}^{M} \sum_{y=1}^{N} (S_{xy} - C_{xy})^2 \quad (7)$$

Here, $c_{\text{max}}$ indicates the maximum value in the image, for example:

$$c_{\text{max}} = \begin{cases} 255 & \text{double precision images} \\ 255 & \text{in 8-bit unsigned integer intensity images} \end{cases}$$

In addition, x and y are the image coordinates, M and N are the dimensions of the image, $S_{xy}$ is the resultant stego image, and $C_{xy}$ is the cover image. In the literatures (Drew & Bergner, 2008; Hashad, Madani, & Wahdan, 2005; Kermani & Jamzad, 2005; X. Li & Wang, 2007; Y.-H. Yu, Chang, & Lin, 2007), $c_{\text{max}}$ is set to 255, as a default value for 8-bit images. It can be that an image has only up to 253, or fewer, gray colors. Knowing that $c_{\text{max}}$ is raised to the power of 2 results in a severe change to the PSNR value. For this reason, $c_{\text{max}}$ is considered as the actual maximum value rather than the largest possible value. PSNR is often expressed on a logarithmic scale in decibels (dB). PSNR values below 30 dB indicate low quality (i.e., distortion caused by embedding is clear). A high-quality stego image should strive for a PSNR of 40 dB, or higher (Hernandez-Castro, Blasco-Lopez, Estevez-Tapiador, & Ribagorda-Garnacho, 2006; Zamani, Manaif, & Abdullah, 2012).

Evaluation of the Mentioned Techniques:

All the above mentioned algorithms with respect to image steganography are not void of weak and strong points. Consequently, it is important to decide the most suitable approach to be applied. As defined before, there are three main parameters to measure the performance of the steganographic system (Amirtharajan & Rayappan, 2012). Figure 7 shows the relationship between steganography parameters (Fridrich, 1999). These parameters are as follows:

- Undetectability (imperceptibility): this parameter is the first and the primary requirement; it represents the ability to avoid detection, i.e., where the human eye fail to notice it. However, the techniques that do not alter the image in such a way to be perceptible to the human eye may still alter the image in a way that it is detectable by the statistical tests. Truly secure steganographic techniques should be undetectable neither by the human eye nor by the statistical attacks (Amirtharajan & Rayappan, 2012; Bahi, Couchot, & Guyeux, 2012).
Robustness: it is the second parameter that measures the ability of the steganographic technique to survive the attempts of removing the hidden information. Such attempts include, image manipulation (like cropping or rotating), data compression, and image filtering (Bahi et al., 2012). Watermarks are an example of a robust steganographic technique.

Payload capacity: it is the third parameter that represents the maximum amount of information that can be hidden and retrieved successfully. When compared with watermarking, that requires embedding only a small amount of copyright information, steganography is seen to hide communication and consequently a sufficient embedding capacity is required. Accordingly and by using this parameter, small amounts of data could be hidden without being detected by the human eye. Larger amounts of information, on the other hand, may detect artifacts by the HVS or statistical tests (G. Chen, Zhang, Chen, Fu, & Wu, 2012; Lee & Huang, 2012).

Fig. 7: Competing factors in steganographic systems.

The following paragraphs compare the previously mentioned steganographic techniques in terms of the three competing parameters.

LSB technique in the spatial domain is a practical way to conceal information but, at the same time, it is vulnerable to small changes resulting from image processing or lossy compression (N. F. Johnson & S. Jajodia, 1998). Although LSB techniques can hide large quantities of information i.e., high payload capacity, they often compromise the statistical properties of the image and thus indicate a low robustness against statistical attacks as well as image manipulation.

The promising techniques such as DCT, DWT and the adaptive steganography are not tended to attacks, especially when the hidden message is small. This can be justified in relation to the way they change the coefficients in the transform domain. Generally speaking, such techniques tend to have a lower payload when they are compared to the spatial domain algorithms (Cheddad et al., 2010). The experiments on the (DCT) coefficients have introduced some promising results and then they have diverted the researchers’ attention towards JPEG images. Working at some level like that of DCT turns steganography much more powerful and less prone to statistical attacks. Embedding in the DWT domain reveals a sort of constructive results and outperforms DCT embedding, especially in terms of compression survival (Cheddad, 2009).

Spread spectrum techniques are generally quite robust against statistical attacks, since the hidden message is spread throughout the image. Spread spectrum encoding is extensively used in military communications due to its robustness against detection. When a message is embedded, an attacker cannot be easily recognized and it will be difficult to extract it without knowing the suitable keys. SISS is very good for steganography because of the reasonable high capacity and high difficulty proposed in the process of detection and extraction. Anyway, this method is still vulnerable to destruction from compression and image processing. A determined attacker can quite easily compromise the embedded data using some digital processing, like for example noise reduction filters, the same that are used in decoding to estimate the original cover. Moreover, the basic tradeoff in using SSIS is between the error rate could be afforded and the amount of information to be embedded, that varies in turn the power of the added noise. The used error correcting coding (ECC) must be chosen carefully to allow for a low power without increasing the BER as well.

The statistical techniques in most cases are vulnerable to cropping, rotating, and scaling attacks, along with any attacks that work against the watermarking technique. Defenses could be considered to make the statistical techniques as robust as the watermarking scheme as detailed in (section D). The payload capacity and invisibility depends on the cover image selected.

Unlike many LSB methods, distortion techniques do not upset any statistical properties of the image. In contrast, the need to send the cover image over a secure channel limits the worth of this technique. As in any steganographic technique, the cover image should never be used more than one time. If an attacker alters the stego-image by cropping, rotating, or scaling, the alteration can easily be perceived by the receiver and can fairly be reversed to the point where the message encoded with error correcting information can be fully
recovered. Error correcting information also aids if the stego-image is filtered through a lossy compression scheme such as JPEG. Adopting this technique limits the hidden information capacity, since adding distortion to the cover image is the basis of embedding algorithm. As a result, the distorted image will be more vulnerable to the HVS.

Techniques that modify image file formatting information have a large payload; however they have the following drawbacks: they are easily detected and defeated; they are not robust against lossy compression and image filters, and the issue of saving the image one more time totally breaks the hidden data (Cheddad et al., 2008b).

Hiding information via steganographic techniques that modify the elements in the visual image results in a stego-image that will survive rotation, scaling and much lossy compression like JPEG. A reasonable payload capacity can be achieved with this technique as well. Table 1 summarizes the evaluation of the mentioned techniques through this paper.

**Table 1: A comparison of image steganography techniques**

<table>
<thead>
<tr>
<th></th>
<th>LSB</th>
<th>Transform Domain</th>
<th>Spread Spectrum</th>
<th>Statistical Techniques</th>
<th>Distortion Techniques</th>
<th>File and Pallet Embedding</th>
</tr>
</thead>
<tbody>
<tr>
<td>Imperceptibility</td>
<td>High*</td>
<td>High</td>
<td>High</td>
<td>Medium*</td>
<td>Low</td>
<td>High*</td>
</tr>
<tr>
<td>Robustness</td>
<td>Low</td>
<td>High</td>
<td>Medium</td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Payload Capacity</td>
<td>High</td>
<td>Low</td>
<td>High</td>
<td>Low*</td>
<td>Low</td>
<td>High</td>
</tr>
</tbody>
</table>

*: Indicates dependency on the used cover image

**Steganalysis:**

The goal of steganography is to stay away from drawing attention to the transmission of a secret message. If the transmission is suspicious, then this goal is defeated. Steganalysis is the art and science that strives to detect the hidden information within a cover file. An investigator, known as a steganalyst, applies steganalysis to a digital file in an attempt to detect and extract the stego data (M.-C. Chen, 2010). Initially, the science of steganalysis was aimed to detect or estimate the presence of the hidden information based on data transfer observation, without having supposition of the adopted steganography algorithm. In digital image steganalysis an analyst has three goals, determine if an embedded message exists, determine the used embedding method to construct the stego image, and finally, extract the hidden message (Rodriguez, 2008).

Steganalysis is achieved through applying different image processing techniques, e.g., image filtering, rotating, cropping, and translating. Furthermore, it can be done by adopting a coding program to check the stego-image structure and measure its statistical properties. Statistical properties measurements include first order statistics (histograms) or second order statistics (correlation between pixels, direction, and distance). In addition, JPEG double compression and the distribution of DCT coefficients can be considered as a good indication to the use of DCT-based image steganography (Cheddad et al., 2010).

Kharrazi in his study (Kharrazi, 2006), found that one should realize that steganalysis algorithms are considered successful if they are able to detect the existence of the message, and not that important to be able to decode the message itself. The latter would be so difficult if the message is encrypted with one of the strong cryptography algorithms. Nevertheless, in recent times there are specific methods which can estimate the size of the embedded message in addition to detecting the presence of the secret message. Johnson and Jajodia (N. Johnson & S. Jajodia, 1998) introduced the following definitions which are adopted by the steganalysis community:

- **Stego-only attack**: The stego file is the only item available for analysis.
- **Known cover attack**: The original cover and stego file are both available for analysis.
- **Known message attack**: Sometime, the attacker may know the hidden message. Analyzing the stego image for patterns that correspond to the hidden message may be helpful to attack the system in the future. Even with the message, this may be very hard and may even be considered equivalent to the stego-only attack.
- **Chosen stego attack**: The stego file and the used tool (algorithm) are both known.
- **Chosen stego message attack**: The steganalyst generates stego files from a known steganography tool using a chosen stego message. The idea behind this attack is to determine the corresponding patterns in the stego image that may point to the use of particular steganography tools or algorithms.
- **Known stego attack**: The cover file, stego file, and stego tool are known.

**Discussion and conclusion:**

This paper reviewed the main steganographic techniques for both lossy and lossless image formats, such as JPEG and BMP. The consequences are presented in terms of a taxonomy that focuses on three principal...
steganographic techniques for hiding information in image files. Those techniques include those modifying the image in the spatial domain, in the transform domain, and those modifying the image file formatting. Each of these techniques tries to satisfy the three most important factors of steganographic design (imperceptibility or undetectability, capacity, and robustness). From Table 1, one can deduce that while one technique may lack in payload capacity, another may lack in robustness. For example, file formatting techniques can store large amounts of information, but they are easily detected and attacked. Likewise, LSB techniques in a spatial domain have a high payload capacity, but they often fail to prevent statistical attacks and are thus easily detected. It is important to notice that the hiding capacity in LSB technique depends on the cover image being used. LSB in BMP images is capable of hiding relatively a large message, but large amount of altered bits results in a larger possibility of detection by human eye. While LSB in GIF images is approximately the same as that of using LSB in BMP images. The only difference is related to the structure of the GIF images, since they only have a bit depth of 8. Thus, the amount of hidden information is less than with BMP. In addition, LSB in GIF is mainly dependent on the file format and the image itself. Incorrect choice of cover image could result in visible message.

Besides, file and spatial domain approaches are considered not to be robust against lossy compression and filtering. Transform domain techniques are considered more robust for lossy compression image formats, but this advantage is achieved at the expense of payload capacity.

However, it is possible to defeat the transform domain techniques, but with some efforts. For most of steganography applications, JPEG file format can be used, especially for images that have to be communicated over an open systems environment like the Internet. Thus, for an agent to send secret information using steganographic techniques, he or she must select a suitable steganographic algorithm and suitable cover image as well. The required application is the only thing to decide the most appropriate steganographic method among all the present image steganographic techniques. In short, one must have the determination to compromise on some characteristics to ensure the high performance of other characteristics.

REFERENCES


