ISSN No. 0976-5697

Volume 3, No. 7, Nov-Dec 2012

International Journal of Advanced Research in Computer Science

REVIEW ARTICLE

Available Online at www.ijarcs.info

Steganography for Invisible Communication: A Review

Awdhesh Kumar Shukla*, Scientist
Knowledge Dissemination Group
CSIR-Central Scientific Instruments Organisation
Chandigarh, India
akshukla@csio.res.in, shukla_aks2001@yahoo.com

Vishu
DAV Institute of Engineering and Technology
Jalandhar, India
v_aeri313@yahoo.co.in

Amod Kumar, Chief Scientist
Head, Bio-Medical Instrumentation
CSIR-Central Scientific Instruments Organisation, Chandigarh, India
amod@csio.res.in, csioamod@yahoo.com

Abstract: Steganography is the science and art of embedding secret messages in innocuous looking carriers in such a way that it does not draw the attention of anyone other than the sender and the targeted recipient, thus a method for secret and invisible communication which provides security through obscurity. Its main purpose is to hide the occurrence of communication over a public channel. Steganography has been used since ancient times and has grown exponentially in the recent past because of the improvements in computing power. Earlier, steganography was implemented using some physical medium i.e. some tangible objects but now a days, it is implemented electronically by using several other intangible objects i.e. data can be hidden using any type of media, be it image in bmp, jpeg, gif format or some music file, video clip, text file, SMS etc. In this paper, different types of techniques used to hide data have been discussed with major focus on image based modern steganographic techniques.

Keywords: Steganography, steganalysis, cryptography, digital watermarking, stego-key

I. INTRODUCTION

Steganography is a process or technique that enables a user to transmit some secret data over a communication media after embedding it behind some cover. Unlike watermarking and cryptography, the main focus in steganography revolves around concealing the existence of any secret communication taking place.

A. Steganography Vs. Cryptography:

The basic purpose of both the steganography and cryptography is same i.e. to ensure secret communication. However, steganography is not the same as cryptography. Basic differences between the two are:

a. Steganography is hidden writing. In steganography, only the sender and the receiver know the existence of the message. Although the message is there, but nobody else notices it. However, once noticed, it can be read and manipulated. On the other hand, cryptography is secret writing, anyone can see the message, but no one can read it. This is because its letters have been re-arranged, or substituted by different letters, according to some scheme that only the sender and receiver know.

b. Steganography uses methods that would hide both the message and save the contents while Cryptographic methods protect the contents of a file.

A combination of steganography and cryptography can provide improved communication security.

Let us consider the example of a ring. To hide the ring in a house to save it from theft, one can place it in a safe and then lock it using a key. This is cryptography. However, if the ring is placed behind a common object i.e. book or anything, the thief can't probably think that the ring is hidden at such obvious place. This is steganography. Placing the ring in locker and then hiding the locker behind the wall is steganography in combination with cryptography.

B. Steganography vs. Digital Watermarking:

The process of embedding information into digital document in a manner such that the embedded information may be used to verify the identity or authenticity of the owner is known as Digital Watermarking, similar to that of paper bearing watermarks. Relation between Steganography and Digital Watermarking can be described as:

c. The main goal of steganography is to hide a message in cover medium to obtain a new data file, practically indistinguishable from the cover medium in such a way that an eavesdropper does not doubt the presence of message there. Watermarking is to hide a message in cover medium to obtain a new data file, practically indistinguishable from the cover medium in such a way that an eavesdropper can see the message but is not able to remove or replace its contents.

d. Steganography hides messages in one-to-one communications and Watermarking hides messages in one-to-many communications.

e. Security is not a concern in steganography i.e. providing protection against removing or modification of the hidden message is not a major issue. Only data embedding is the main issue. On the other hand, watermarking methods are robust, in nature, to attempts to remove or modify the hidden message.

II. STEGANOGRAPHY IN VARIOUS AGES

Steganography is not a modern technique of data hiding as it has its roots in the past. Some of the ways used in an-

cient times to send some secret data have been categorized under Physical forms of steganography as below:

A. Physical Forms:

Before the evolution of the computer system, messages were hidden on the tangible or physical objects. Several data hiding methods, used by different countries, have been reported in the literature.

Gaspar Schott (1608-1666) wrote the book named as "Schola Steganographica," in which the technique to hide messages in music scores has been discussed. Here, each letter of the message corresponds to one note [1]. The 'Ave Maria' code, originally proposed by Johannes Trithemius (1462-1516), is also expanded in this book in forty tables and each table consists of twenty four entries in four languages. Each letter was replaced by the word in the corresponding table entry and thus the stego text was prepared. It has been discovered that these forty tables can be translated by reducing them modulo 25 and applying reverse alphabets [2]. Another method, used by J.S. Bach, makes use of music scores to hide data on the basis of the number of occurrences of notes [3]. John Wilkins (1614-1672) described how two musicians can communicate with each other by playing the instruments of music and also by talking with the instruments of speech [4]. The art of hiding message in the geometric drawings using points, ends of lines etc. has also been covered in [4]. The use of acrostic to hide message has been discussed in the book named as "The Codebreakers," by David Kahn [5]. This book elaborates how a monk put his lover's name in the very first letter of the successive chapters of a book. The use of Ciphers for hatching "The Babington Plot" in March 1586 to assassinate Queen Elizabeth and put Mary, Queen of Scots, a Catholic, on the English throne [6] led to the imprisonment and subsequent execution of Mary.

Besides nonliving objects, human factors also contributed in steganography. In 5th century BC, Histaiacus shaved the head of a messenger, then wrote the secret message on his bald head and waited for hair to grow back to send the messenger to the other party [7]. To retrieve message, his hairs were shaved again. Obviously, this method was very time consuming. Hand positions can also be used to form some sequence of the message e.g. during Viet-Namm war, the captured crew members of the U.S. armed forces used hand positions during photos to be guessed by the media [8]. Several steganographic techniques had been used during World War II. Nazis developed microdots, the microfilm chips created at high magnification and usually of the size of periods, which could contain large information [9]. In another method, a security protocol was developed by ancient China in which the sender and the receiver had same paper mask having a number of holes at random locations. The sender could write the secret message into the holes by placing his mask over a paper, remove the mask and compose a cover message.

The receiver could get the secret message by placing his mask over the letter received. Cardano Grill [10], a device invented by Girolama Cardano (1501-1576), was used to retrieve data. The grill is to be placed on received printed text and the intended message is observed. Other such techniques may be overwriting printed texts or pin puntures in texts. The above method seems to be reinvented by a British Bank which evolved the system that its customers conceal the personal identification number (PIN) used with their ATM card but poor implementation weakened the system [11]. There were various methods that helped slaves to aid in their escape. One such method was using various patterns in quilts, commonly hung from windowsills to dry. One such quilt pattern could be a sailboat symbol which signifies that either a water body was nearby or boats were available to escape [12].

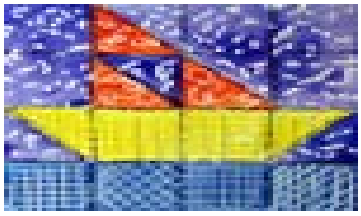

Figure 1: Sailboat Quilt Code [12]

Writing on the wood beneath a tablet and then covering it with wax, was the method used by Demeratus [6], a Greek at the Persian court. In the technique invented by Aeneas, data was hidden by making holes either above or below the letters in the cover text. These small holes were hidden by contrasting between black letters and white papers. This technique was in use in 17th century and later on, Wilkins improved it by using invisible ink to print very small dots instead of making holes [4]. German spies also used this technique during world wars [5]. Invisible ink, made up of organic substances i.e. milk or salt armoniack dissolved in water and developed heat which helped in steganography, but this technology did not succeed because of the invention of "universal developers". These could determine the parts of paper being wetted from the effects on the surface of fibers [5]. Other common techniques of data hiding include letters hidden in messengers' soles, women's ear-rings, notes carried by pigeons etc.

B. Steganography in Modern/Present Form:

In the modern era, computer system has become backbone of all the great work. The use of computer system has left a good impact on steganography as well. The reason is that many cover media such as images, audio, video, text etc can be used and manipulated digitally to perform steganography. Moreover, a natural cover medium for steganography - the human DNA strand itself – has been unearthed by science.

A steganographic system consists of two components: Encoder and Decoder.

a) ***Encoder:*** The encoder is the main component of the steganographic system. The "secret message" can be defined as the data that is required to remain confidential. The "cover" is the medium in which the message is embedded and which serves to hide the presence of the message. The "stego-image" (when an image is used as cover) carries the secret message embeded within itself by using an algorithm ("Secret Key"). The encoder embeds the secret message beneath cover medium. Encoder is programmed on the basis of some data hiding algorithms. The data is hidden at the redundant places of the cover file e.g. an image as the changes made in such regions are not easily detectable. Fig.2 describes the basic operation of an encoder.

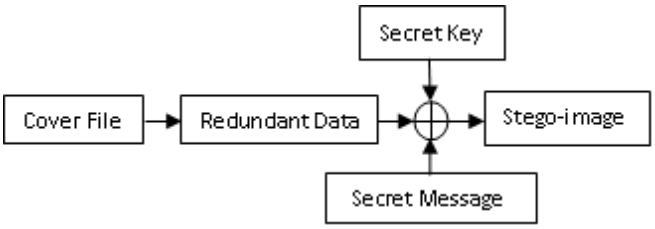

Figure.2: Basic Functioning of an Encoder

The secret key tells the locations of the regions of the cover image that have been replaced with the secret message. This key is used at extraction stage. The size of the hidden message must be less than or equal to the size of redundant data available for encoding otherwise the encoder would not be able to hide all the data.

b) ***Decoder:*** The function of decoder is opposite to that of the encoder. It takes a stego file, makes use of a secret shared key and on the basis of certain algorithms extracts secret data. In case of image steganography, the exact replica of original hidden image can't be reproduced. Fig.3 represents the basic functioning of a decoder.

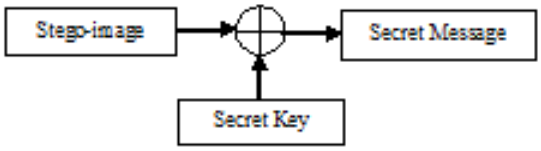

Figure.3: Basic Functioning of a Decoder

III. IMAGE STEGANOGRAPHY

Image Steganography is the most popular type of steganography. The scope of image steganography is large because of the various image formats available such as BMP, JPEG, PNG, GIF etc. The user can opt from one of these image formats as required. Different steganographic techniques have been developed on the basis of these different image formats. Before discussing these techniques, let us discuss about image and available formats in brief.

A. ***Image:***

In general, a digital image is an arrangement of small dots known as pixels (picture elements), each having different light intensity [13]. The bit depth is the number of bits in a pixel. The smallest bit depth for colour images is 8, which means 8 bits are used to describe the color of each pixel [14]. Thus, 8-bit depth color and grayscale images can display 256 (i.e. $\mathbf{2^{bit\ depth}}$) different colors or shades of grey respectively. A 24-bit color image can display upto 16,777,216 (2^{24}) discrete combinations of Red, Green and Blue values. These images use RGB color model which is also known as true color model. Here, every 8-bits of 24 bits represent one of the three color components i.e. red, green and blue.

B. ***Digital Image Formats:***

a. ***GIF:***

GIF, short for Graphics Interchange Format is a bitmap image format, introduced in late eighties which has come into widespread usage for webpages. The GIF format supports up to 8-bits per pixel, thus allowing a single image to reference a palette of up to 256 distinct colors, chosen from the 24-bit RGB color space.

b. ***BMP:***

This image format is also known as bitmap image file format common for MS Windows. The BMP images are large in size and their quality varies from medium to high.

c. ***JPEG:***

The term "JPEG" is an abbreviation for the Joint Photographic Experts Group. JPEG is a commonly used file format of lossy compression for digital image. This is the most widely used image format for photographic images. JPEG images are of high quality and small in size.

C. ***Basic Techniques of Steganography:***

Image Steganography techniques can be divided into three categories which are described as follows:

a. ***Least significant bit method [15]:*** These methods hide the most significant bits of secret message in the least significant bits of the carrier (cover) image. This method is also known as LSB method. Example: consider three pixels of 24 bit image with bit values as:

01010110 11001010 10101000
00011010 11010101 01010011
10101110 01011110 11011000

These are the pixel values of the cover image. Now suppose we want to hide secret message 01010100 in these bits. Starting from the most significant bit of the secret message, we embed each of these bits in the LSB of the cover medium. This results in the pattern:

01010110 11001011 10101000
00011011 11010100 01010011
10101110 01011110 11011000

Here, the highlighted bits are the changed ones. Though this technique is very simple, it suffers from certain limitations. Embedding secret data requires large cover images and if the secret message is compressed using a lossy algorithm, then at the extraction stage, the extracted message may not be the correct one.

b. ***Masking and Filtering:*** Here the data is embedded by changing the intensity of pixels of the image. The luminance properties of the image are varied so that human eye is not able to notice any change. This method is more robust than LSB in many ways like compression, cropping and various image processing as it only uses the visual aspects of the cover image. Moreover, as data is hidden in the visible parts of the cover image instead of noisy regions, it is better than LSB in lossy compression algorithms.

c. ***Transformations:*** the message is hidden behind the cover image by modulating coefficients in transfer domain such as DCT, DFT or wavelet transform.

D. Various Data Embedding Techniques based on Transformations:

a. ***Jsteg [16]:***

This algorithm was developed by Derek Upham. It is resistant against the visual attacks [17]. Moreover, it offers a very good capacity for steganographic messages (e. g. 12.8% of the steganogram's size). After quantization, this algorithm skips all coefficients of value 0 or 1 and replaces the LSB's of the rest of the frequency coefficients by the secret message [16].

Andreas Westfeld and Andreas Pfitzmann noticed that changing the LSBs sequentially results in distortion due to which a steganalyst can easily doubt on the occurrence of some hidden message[17]. They also observed that embedding high entropy data can cause a visible change in the histogram of the color frequencies.

b. F3:

Contrary to Jsteg, F3 makes use of coefficients having value 1. It decrements the coefficient's absolute values if their LSB does not match—except coefficients having value 0, because the absolute value can't be decremented in this case. That's why zero coefficients are not used steganographically. After embedding, the LSB of non-zero coefficients match the secret message, but bits have not been overwritten, as Chi-square test can easily detect all such changes [17].

Main flaw in F3 technique is that several embedded bits become victim to shrinkage which occurs when F3 decrements the absolute values of 1 and −1 resulting in a 0. Distinguishing a zero coefficient, which is logically unused, from a 0 produced by shrinkage becomes impossible for the receiver. Thus all the zero coefficients are skipped. So, the sender embeds the affected bit again and again as he notices when he produces a zero.

Shrinkage arises only when we embed a zero bit. The repetition of zero bits shifts the originally equalized ratio of steganographic values in favor of the steganographically produced 0s.

F3 algorithm produces a good number of even coefficients as steganographic zeroes. Hence, this process produces more even coefficients than odd.

c. F4:

F4 algorithm eliminates the shortcomings of F3 by mapping negative coefficients to the inverted steganographic values i.e. even negative coefficients represent a one (steganographically produced), odd negative a zero, even positive represent a zero (same as with Jsteg and F3), and odd positive a one.

d. F5 [18]:

F5 algorithm selects DCT coefficients randomly to embed secret data bits. Thereafter, it applies matrix embeddin, due to which the changes needed on the cover image to embed secret bits get reduced.

a) Embedding procedure:

i. The RGB values of the cover image are obtained.

ii. The quantization table is prepared using a quality ratio q, the image is compressed and we obtain quantized DCT coefficients.

iii. The embedding capacity,say C, of the cover image is computed using the formula

$C=h_{DCT}-h_{DCT}/64-h(0)-h(1)+0.49h(1)$

Where h_{DCT} is the number of all DCT coeffecients.

h(0) is the number of AC coefficients with value zero.

h(1) is the number of AC coefficients with value one.

iv. For the security purposes, user is required to enter a password on the basis of which random bits of the cover image are chosen to embed secret data. Moreover, on the basis of the password, a seed is generated that serves as the initial point to generate the pseudo-random bit stream. This bit stream is further XOR-ed with the message bit to make it random.

v. The k bits of the message are embedded into 2^k-1 coefficients randomly. If the hash of the coefficients does not match with the secret bits, then one of the coefficients is decremented by one. After decrement, if the coefficient becomes zero, shrinkage occurs, due to which the same secret bits will be embedded in the next 2^k-1 coefficients.

This algorithm is not traceable by chi-square test as no bit is being replaced here.

To embed two bits say *p, q* in three modifiable bit places *m*1, *m*2, *m*3 changing one place at most, these four cases arise:

p = m1 XOR m3, q = m2 XOR m3	change nothing
p = m1 XOR m3, q = m2 XOR m3	change m1
p = m1 XOR m3, q = m2 XOR m3	change m2
p = m1 XOR m3, q = m2 XOR m3	change m3.

e. DCT coefficient selection method [19]:

Here the secret data is embedded in jpeg image on the basis of quantization error table, i.e. QET.

a) Embedding Procedure:

Select the coefficients that turn out to be zero after quantization step.

(a). Dequantize these coefficients and let these be f(i,j), where (i,j) represents the jth element of ith row of the quantization matrix.

(b). QET is used to find the number of bits that can be embedded into the selected coefficient. For this, following formula is used :

$N(i,j)= \text{Log}_2(QET(i,j) +1)$

Let m be the secret data to embed and E(i,j) be the quantized DCT matrix after embedding. The secret data is embedded using the formula:

$E(i,j)= f(i,j) + m \quad \text{if } QET(i,j) > 0$

or

$E(i,j)= f(i,j) – m \quad \text{if } QET(i,j) < 0$ ----- (1)

Where $0<=m<=|QET(i,j)|$

The embedded DCT block is coded using some compression method i.e. run-length encoding of Huffman algorithm.

The most important step is to change the entries in the quantization table corresponding to the selected DCT coefficients by 1. It is done to avoid significant distortion in the image that will be reconstructed in extraction process.

b) Extraction procedure:

The quantization matrix is searched to check the coefficients having value equal to 1 because these are the places where secret message is embedded.

a) After original jpeg image is compressed using quantization factors say q1,q2 DCT block is dequantized and the QET is made.Now, we have E(i,j) and f(i,j).

b) Putting these in equation (1) gives the value of m. Such values are collected to form the whole message string.

f. Random number method [20]:

Here random numbers are used to generate the positions in the cover image to embed secret message. Random numbers can be generated using any of the following methods:

a) Type I Method: To get the location of the byte where data is to be embeded next, following general congruential method may be used.

$$x_{i+1}=(a_1 x_i + a_2 x_{i-2} +\ldots.+ a_n x_{i-n+1} +c) \pmod m$$

Where m, n, a, c and x are non-negative integers. x_i is the previous random number and $x_i +1$ is the place where bit of secret message is to be placed next.

The numbers generated by this method are between 0 and m-1. e.g. consider that we have three adjacent pixels with the following RGB encoding:

10010101 00001101 11001001
10010110 00001111 11001010
10011111 00010000 11001011

Now suppose we want to hide the data : 1011.

Random number generator used for hiding data is:

$$x_{i+1} = ax_i + c \pmod m$$

Using above formula with values x_0= a= c= 7 and m = 3

X1 = (7*1 + 7) mod 3
= 14%3
= 2

X2 = (7*2 + 7) mod 3
= 21%3
= 0

X3 = (7*0 +7) mod 3
= 7%3
= 1

X4 = (7*1 + 7) mod 3
= 56%3
= 2

X5= (7*2 + 7) mod 3
= 21%3
= 0

X6 = (7*0 +7) mod 3
= 7%3
= 1

The repeated pattern of 2,0,1,2,0,1,….. is generated which means the message can be embedded at 2nd , 0th and 1st positions of the given binary string. The result is given as:

10010101 00001101 11001001
10010110 00001111 11001011
10011111 00010000 11001011

Bold bits represent the hidden data and underlined bit represents modified bits.

b) Type II methods: Here, we combine two or more Type I methods and in addition, there is a control procedure used to control the sequence of these methods randomly. The output sequence of previous method is considered to determine the randomness.

c) Type III Methods: Here, message to be hidden is first encrypted using different encryption algorithms. Then the hidden message is embedded into the cover image through type II method. Encryption key is either randomized or user specific.

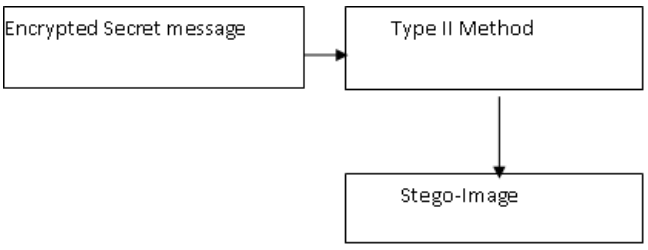

Figure.4: Basic Concept of Type III method

d) Type IV methods: Here, one set of type III method is considered as a block and many such different blocks are combined in parallel, in series or as a combination of the both i.e. hybrid :

a. Parallel combination: cover image and hidden image are divided into the blocks and these blocks of cover image are used to hide the blocks of hidden image. It is suitable for larger images. The data is distributed randomly on the cover image.
b. Series combination: this method is suitable for small sized images. Here, three types of images are to be used: used-cover image, intermediate cover image and hidden image. Secret data is hidden into intermediate image which is further hidden into the cover image to generate the final stego image.
c. Hybrid combination: here, the above two methods are combined to hide the data.

g. SSLDC [21]:

SSLDC stands for Secured Steganography using LSB, DCT and Compression [21]. This technique uses different bits of cover image to perform LSB. The procedure is as below:

a) Apply LSB algorithm on both cover image and hidden image to generate a stego-image. This is the phase where this technique differs from other methods. Four different types of LSB transformations i.e. L1, L2-L5, L6, L7 have been used here.

i. L1 transformation: here first four MSBs of hidden image are hidden into last four LSBs of the cover image. It uses only one byte of cover image, so it is a lossy technique.

ii. L2, L3, L4, L5 transformations: here, two bytes of cover image are used for embedding. In case of L2 transformation, seven MSBs of the secret image are embedded in the seven LSB bits of first byte of the cover image and the last bit of the hidden image is embedded into the last bit of the second byte of the cover image. In a similar manner, L3 uses six LSBs of the first byte and two LSBs of the second byte of the cover image to hide one byte of the hidden image. L4 transformation uses five LSBs of the first byte and three LSBs of the second byte of the cover image to hide one byte of the hidden image. And so on.

iii. L6 Transformation: here, four bytes of the cover image embed one byte of hidden data. Each byte of ***cover*** image replaces its two LSBs with the two MSBs of the hidden data.

iv. ***L7 Transformation:*** here, each LSB of eight bytes of the cover image is replaced by one bit of hidden data.

b) Then DCT is performed followed by quantization and run-length coding on the stego-image to get final stego-image.

c) The reverse procedure is applied on the stego-image to retrieve the hidden image.

h. BPCS Steganography [22]:

In Bit-Plane Complexity Segmentation (BPCS) Steganography, some specific parts of a cover image, such as complex or noisy regions, can be used to hide data because our eyes are not sensitive to detect the slight alterations of such parts of an image. This fact forms the basis of BPCS Steganography [22]. Procedure of BPCS Steganography is:

i. Segment each bit-plane of the cover image into 8*8 block and distinguish between informative and noisy regions on the basis of some threshold value.
ii. Divide the secret message into set of blocks each containing eight bytes of secret data.
iii. Now, the secret file can be simple or complex. If it is simple, noticeable changes can be felt on the cover image. To avoid this, secret file should be converted into the stream of complex blocks and for this purpose, an operation known as conjugation is applied to the simple blocks [23]. Next prepare a conjugation map that specifies which blocks of the secret file are conjugated and this map is also embedded along the secret file as blocks.
iv. Then replace the noisy regions of the cover file with the secret data. Record the conjugated blocks in a conjugation map.
v. Just as the secret data, conjugation map is also embedded in the cover image because it will be useful in the extraction process.

(a). Determining if a block is complex or not: To determine if a block has complex patterns, consider a factor named as black-and-white border complexity measure which is defined as total length of black and white borders within a block. If its value exceeds a threshold, it means the block is a complex one else it is not.

(b). Flaws in BPCS steganography method:

i. It embeds data by replacing complex blocks of cover image with the blocks of the resource file. The black and white border complexity measure, though, is a good parameter to determine complex blocks, but in some cases, it produces false results such as in chess boards. It may consider the whole pattern of chess board as complex one and suitable for embedding.
ii. The blocks at the boundary of the noisy regions and informative regions can be considered as complex ones. If data is embedded in such blocks then the significant changes can be felt on the cover image.

i. ABCDE method[24]:

A new technique called "A Block Complexity Data Embedding (ABCDE)", based on the BPCS method overcomes these drawbacks and uses two new complexity measures to differentiate between complex blocks and simpler ones. These measures are:

a) Run-length irregularity: it is computed on the basis of black and white pixels distributed in rows and columns of a block. It prevents the blocks having periodical patterns to be chosen for embedding.

b) Border noisiness: it is computed on the basis of black and white borders distributed along adjacent rows and columns.

Both these measures can be used simultaneously.

After computing these measures, their values are compared with threshold and if they are larger, then the block is a complex one. The threshold can be specified separately for each block.

IV. TEXTUAL STEGANOGRAPHY

Using text documents as a cover medium to perform steganography is not a new concept. In the 1980s, in Britain, to trace press leaks of the cabinet documents, government word processors were altered to encode a specific user identity in the spaces between words by the British Government during the tenure of British Prime Minister Margaret Thatcher. After the leaked documents were recovered and analyzed, the pattern of spaces were instrumental in establishing the identity of the leaker [25]. Various methods of text steganography [26] are as under :

A. Line Shifting:

Here, the lines of the text are vertically shifted to some extent i.e. each line is shifted 1/300 inch up or down and information is hidden in this space by creating a unique shape of the text. This method is useful for printed texts.

B. Word Shifting:

Here, the words are shifted horizontally and data is hidden in that space. This method can be identified with difficulty as sometimes, to fill a line, distance between words can be varied in natural way.

Both the above methods suffer from following limitations:

a. The distance can be determined using special distance assessment instruments and data can be destroyed by introducing changes.
b. If the text is retyped or some OCR is used, the hidden data gets destroyed.

C. Semantic Methods:

The secret data can be sent by replacing some of its words with their synonyms.

D. Abbreviation:

In this approach, appropriate abbreviations are used corresponding to secret text.

E. Feature Coding:

Here, certain characters such as h, p, d etc. are used to hide data by shortening or elongating their end parts. This method suffers with the problem of OCR programs and retyping the text.

F. Open Spaces:

This method adds extra white spaces to the text and hides information in these spaces. The hidden data can be destroyed as some text editors automatically delete extra white-spaces.

G. Persian/Arabic Text Steganography:

There are lots of points in the characters of Persian/Arabic language. Such points can be used to hide data. This method can also be used in English language in case of characters i and j.

V. DNA BASED STEGANOGRAPHY

The concept of DNA-based steganography is as new as advances in DNA handling techniques i.e. DNA generation, DNA sequencing and related DNA-based techniques [27]. Huffman, comma and alternating codes were developed for DNA based encryption [28]. DNA has exceptionally high data density which means it can store large amount of data in it per unit of mass. So, the DNA molecules can fulfill the high data volume requirement for secured steganography. DNA was proposed as a medium with ultra high storage

density for computational purposes [29]. Afterwards, several other applications have also been demonstrated [30-32]. A DNA based information storage method was reported [33] which addresses complete extended ASCII character set in terms of DNA sequences, thereby claiming to represent all kinds of digital information in terms of DNA sequence. Although DNA is a known and proven medium of information storage, but there is not much literature available on its practical use for steganography, the prime reason being the limitations of DNA technology. The secret information is encoded in the sequence of the DNA strand and flanked by the two secret primer target regions. The encoded DNA strand is hidden amongst a very large amount of similarly sized background DNA strand. The receiving party should know the sequence of primers that binds to the target regions on the message containing DNA strand to extract the message by selectively amplifying that DNA molecule through the Polymerse chain reaction (PCR).

VI. STEGANALYSIS

Steganalysis is the technique to detect the presence of some secret message behind the suspected cover medium. It can be said that steganalysis breaks the purpose of steganography by detecting the existence of some hidden message and thereafter destroying it. Steganalysis can be classified in two ways [34]:

a. Signature Steganalysis
b. Statistical Steganalysis

A. Signature Steganalysis:

The properties of cover file change when some secret message is embedded behind it. Due to such changes, unusual patterns can be detected in the cover file and these patterns are known as signatures. Signature steganalysis looks for such patterns to conclude the existence of hidden message. Though steganography techniques hide content behind cover file in a way so as to remain un-noticeable to the human eye, but due to such signatures, user can suspect some cover file even through naked eyes.

B. Statistical Steganalysis:

Various statistics of the cover file also change when some data is hidden behind it. Statistical steganalysis involves mathematical computations to detect the presence of hidden information.

As mathematical observations are more accurate than visual observations, this method is more powerful than the signature steganalysis.

VII. STEGANOGRAPHY AND STEGANALYSIS TOOLS

Many steganography and steganalysis tools are available on Internet for free. Some of these tools are as follows:

A. Ez-stego:

This is an implementation of steganography in java. It can be downloaded from URL: http://www.stego.com

B. MP3Stego:

It hides secret data behind mp3 files. The data is compressed, encrypted nad then hidden behind the mp3 audio file. It san be downloaded for free from URL:

http://www.petitcolas.net/fabien/steganography/mp3stego/

C. Quick Stego:

It makes use of image files as cover medium to embed secret data. It is available at: http://quickcrypto.com/free-steganography-software.html

D. Virtual Steganographic Laboratory (VSL):

With the help of this software, data can be hidden using LSB method, with Karhunen-Loeve Transform (KLT) or with F5 algorithm. Popular resources for steganalysis tools for cracking various data hiding techniques are available at: http://stegsecret.sourceforge.net/ and http://www.sarc-wv.com/ etc.

VIII. CONCLUSION

In this article, various aspects of steganography have been covered. These include ancient and modern steganography; various cover media used over time such as physical, DNA based and digital cover media and varied embedding methods. The main emphasis has been laid upon the present day steganography i.e. image steganography.

The methods used for steganography have advanced significantly over the past centuries, especially with the support of advancements in computing power. Steganography is not used very frequently and possibilities are numerous. New techniques to embed messages are being developed rapidly, existing are getting modified while ways to detect embedded messages are also advancing.

Steganography is the basis for many digital watermarking techniques, thus, apart from its use for secret or invisible communication, other important use of steganographic techniques would be in the field of digital watermarking. The digital watermarking can provide a way of tracking and establishing ownership of the digital materials.

VII. ACKWONLEDGMENT

Authors are grateful to Dr. Pawan Kapur, Director CSIR-CSIO for providing facilities required for this study and for ever-encouraging supervision.

VIII. REFERNECES

[1] F. A. P. Petitcolas, R. J. Anderson and M.G. Kuhn, "Information Hiding A Survey," Proc. IEEE, special issue on protection of multimedia content, July 1999, vol. 87(7), pp. 1062-1078.

[2] J. Reeds, "Solved: the ciphers in book III of Trithemius' steganographia," Cryptologia, Oct. 1998, vol. XXII(4), pp. 291-317.

[3] F. L. Bauer, "Decrypted Secrets, Methods and Maxims of Cryptology," Berlin, Heidelberg, Germany: Springer-Verlag, 1997.

[4] J. Wilkins, B. Asbach-Schnitker, "Mercury, Or, The Secret and Swift Messenger,", URL: http://books.google.co.in/books?id=Lc38yWFv3p8C&printsec=frontcovr&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false

[5] D. Kahn, The Codebreakers-The Story of Secret Writing, New York, NewYork, U.S.A.: Scribner, 1996, ISBN 0-684-83130-9.

[6] Mary's Ciphers, URL: http://www.nationalarchives.gov.uk/spies/ciphers/mary/ma1.htm

[7] B. Dunbar, "A detailed look at Steganographic Techniques and their use in an Open-Systems Environment," SANS Institute InfoSec Reading Room, URL: http://www.sans.org/reading_room/whitepapers/covert/detailed-steganographic-techniques-open-systems-environment_677

[8] "What the Returning POWs Said About Missing Men: The Pink, Blue, and White Pages," URL: www.miafacts.org/pages.htm

[9] J. C. Judge, "Steganography: Past, Present, Future," SANS Institute InfoSec Reading Room: GSEC Version 1.2f, URL: http://www.sans.org/reading_room/whitepapers/steganography/steganography-past-present-future_552

[10] Cardan Grille, URL: http://en.wikipedia.org/wiki/Cardan_grille

[11] R. Anderson, "Why cryptosystems fail," Communications of the A.C.M., Nov. 1994, vol. 37(11), pp. 32-40.

[12] Quilt Codes, URL: http://www.osblackhistory.com/quiltcodes.php

[13] N. F. Johnson and S. Jajodia, "Exploring Steganography: Seeing the Unseen," Computer Journal, February 1998, pp. 26-34,

[14] M. Owens, "A discussion of covert channels and steganography", SANS Institute, 2002

[15] T. Morkel, J. H. P. Eloff and M. S. Olivier, "An Overview of Image Steganography," Proc. Fifth Annual Information Security South Africa Conference (ISSA2005), Sandton, South Africa, June/July 2005 (Published electronically). URL: http://mo.co.za/open/stegoverview.pdf

[16] A. Westfeld, "F5—A Steganographic Algorithm , High Capacity Despite Better Steganalysis," Lecture Notes in Computer Science, 2001, vol. 2137/2001, 289-302.

[17] A.Westfeld and A. Pfitzmann, "Attacks on Steganographic Systems," Information Hiding. Third International Workshop, LNCS 1768, Springer-Verlag Berlin Heidelberg 2000. pp. 61–76.

[18] J. Fridrich, M. Goljan and D. Hogea, "Steganalysis of JPEG Images: Breaking the F5 Algorithm," Lecture Notes in Computer Science, 2003, Vol 2578/2003.

[19] Hsien-Wen Tseng and Chin-Chen Chang, "Steganography using JPEG-Compressed Images," Proc. Proceedings of the Fourth International Conference on Computer and Information Technology (CIT '-04), IEEE Computer Society Washington, DC, USA, 2004, pp. 12-17.

[20] S. Manchanda, M. Dave, S. B. Singh, Sanjeev Manchanda, Mayank Dave and S. B. Singh, " Customized and Secure Image Steganography Through Random Numbers Logic," Signal Processing: An International Journal, 2007,vol. 1(1),

[21] K. B. Raja, C. R. Chowdary, K. R. Venugopal, L. M. Patnaik, "A Secure Image Steganography using LSB, DCT and Compression Techniques on Raw Images," Proc. IEEE: 3rd International Conference on Intelligent Sensing and Information Processing, Bangalore, India, December 2005, pp. 170-176.

[22] E. Kawaguchi and R. O. Eason, "Principle and applications of BPCS-Steganography," Proc. SPIE: Multimedia Systems and Applications, 1998, vol.3528, pp. 464–472.

[23] M. Niimi, H. Noda, and E. Kawaguchi, "An image embedding in image by a complexity based region segmentation method," Proc. International Conference on Image Processing, 1997, vol.3, pp. 74–77.

[24] H. Hirohisa, "A Data Embedding Method using BPCS Principle with New Complexity Measures," URL: http://www.i.h.kyoto-u.ac.jp/~hioki/research/DH/files/abcde_steg02_revised.pdf

[25] Vice Over IP: The VoIP Steganography Threat. URL: http:// http://spectrum.ieee.org/telecom/internet/vice-over-ip-the-voip-steganography-threat/0

[26] M. Shirali-Shahreza and M. H. Shirali-Shahreza, "Text Steganography in SMS," Proc. 2007 International Conference on Convergence Information Technology (ICCIT '07), pp. 2260-2265, IEEE Computer Society Washington, DC, USA

[27] C. T. Clelland, V. Risca and C. Bancroft, "Hiding messages in DNA microdots," Nature, 10 June 1999, vol. 399, pp. 533-534.

[28] G. C. Smith, C.C. Fiddes, J. P. Hawkins, J. P. Cox, "Some possible codes for encrypting data in DNA," Biotechnol Lett., 2003, vol.25(14), pp.1125-1130.

[29] L. M. Adleman, "Molecular Computation of Solutions to Combinatorial Problems," Science, Nov. 1994, vol. 266(5187), pp. 1021-1024.

[30] R. J. Lipton, "DNA Solution of Hard Computational Problems," Science, 1995, vol. 268(5210), pp. 542-545.

[31] F. Guarnieri, M. Fliss, C. Bancroft, "Making DNA Add," Science, 1996, vol. 273(5272), pp. 220-223.

[32] Q. Ouyang, P. D. Kaplan, S. Liu, A. Libchaber, "DNA Solution of the Maximal Clique Problem, Science, 1997, vol.278(5337), pp. 446-449.

[33] L. M. Bharadwaj, A. K. Shukla, A.P Bhondekar, R. Kumar and R. P. Bajpai, "Method for storing information in DNA," US Patent Application 20050053968, March, 2005.

[34] A. Nissar, A. H. Mir, "Classification of steganalysis techniques: A study," Digital Signal Processing, 2010, vol. 20(6), pp. 1758–1770.

