ISSN Print: 2394-7500
ISSN Online: 2394-5869
Impact Factor: 5.2
IJAR 2016; 2(11): 144-149
www.allresearchjournal.com
Received: 21-09-2016
Accepted: 22-10-2016

Samir Kumar Bandyopadhyay
Research Director, Techno India University and 2 Chairman, Techno India Group Techno India University, EM-4, Sector-V, Salt Lake, Kolkata-700091, West Bengal, India

Goutam Roy Chowdhury
Research Director, Techno India University and 2 Chairman, Techno India Group Techno India University, EM-4, Sector-V, Salt Lake, Kolkata-700091, West Bengal, India

Correspondence
Samir Kumar Bandyopadhyay
Research Director, Techno India University and 2 Chairman, Techno India Group Techno India University, EM-4, Sector-V, Salt Lake, Kolkata-700091, West Bengal, India

Network steganography with DCT approach

Samir Kumar Bandyopadhyay and Goutam Roy Chowdhury

Abstract
In this paper, the authors propose a new steganography algorithm to hide data inside image using steganography technique. The proposed algorithm uses binary codes and pixels inside an image. The zipped file is used before it is converted to binary codes to maximize the storage of data inside the image. By applying the proposed algorithm, a system called Steganography Imaging System (SIS) is developed. The system is then tested to see the viability of the proposed algorithm. Various sizes of data are stored inside the images and the PSNR (Peak signal-to-noise ratio) is also captured for each of the images tested. Based on the PSNR value of each images, the stego image has a higher PSNR value. This method also uses 3-dimentional discrete cosine transformation scheme, along with 2-dimentional discrete cosine transformation for sending secret data over network. The paper will not consider network protocol facility by assuming data may be transmitted from one place to another through computer network.

Keywords: Steganography algorithm, secret key, image processing, data retrieval

Introduction
This paper proposes a new algorithm to hide the data inside images using steganography technique. An algorithm is designed to hide all the data inputted within the image to protect the privacy of the data. Then, the system is developed based on the new steganography algorithm. This proposed system provides an image platform for user to input image and a text box to insert texts. Once the proposed algorithm is adapted, user can send the stego image to other computer user so that the receiver is able to retrieve and read the data which is hidden in the stego image by using the same proposed system. Thus, the data can be protected without revealing the contents to other people.

Steganography Imaging System (SIS) is a system that is capable of hiding the data inside the image. The system is using 2 layers of security in order to maintain data privacy. Data security is the practice of keeping data protected from corruption and unauthorized access. The focus behind data security is to ensure privacy while protecting personal or corporate data.

Privacy, on the other hand, is the ability of an individual or group to seclude them or information about themselves and thereby reveal them selectively. Data privacy or information privacy is the relationship between collection and dissemination of data, technology, the public expectation of privacy, and the legal issues.

Data privacy issues can arise from a wide range of sources such as healthcare records, criminal justice investigations and proceedings, financial institutions and transactions, biological traits, residence and geographic records and ethnicity. Data security or data privacy has become increasingly important as more and more systems are connected to the Internet. There are information privacy laws that cover the protection of data or information on private individuals from intentional or unintentional disclosure or misuse. Thus, hiding the data in a kind of form such as within an image is vital in order to make sure that security or privacy of the important data is protected.

The rest of the paper is organized as follows. Section 2 reviews the related work and section 3 presents the proposed algorithm. The implementation of the system is discussed in section 4 together with the discussion of various results obtained from testing the system based on the proposed algorithm with various sizes of data. The image is also tested using the PSNR value. Finally, we conclude the paper in section 5.

Related Work

Hiding data is the process of embedding information into digital content without causing perceptual degradation [1]. In data hiding, three famous techniques can be used. They are watermarking, steganography and cryptography. Steganography is defined as covering writing in Greek. It includes any process that deals with data or information within other data. According to Lou *et al*. [2], steganography is hiding the existence of a message by hiding information into various carriers. The major intent is to prevent the detection of hidden information.

Research in steganography technique has been done back in ancient Greek where during that time the ancient Greek practice of tattooing a secret message on the shaved head of a messenger, and letting his hair grow back before sending him through enemy territory where the latency of this communications system was measured in months [3]. The most famous method of traditional steganography technique around 440 B.C. is marking the document with invisible secret ink, like the juice of a lemon to hide information. Another method is to mark selected characters within a document by pinholes and to generate a pattern or signature [3]. However, the majority of the development and use of computerized steganography only occurred in year 2000 [4]. The main advantage of steganography algorithm is because of its simple security mechanism. Because the steganographic message is integrated invisibly and covered inside other harmless sources, it is very difficult to detect the message without knowing the existence and the appropriate encoding scheme [5]. There are several steganography techniques used for hiding data such as batch steganography, permutation steganography, least significant bits (LSB), bit-plane complexity segmentation (BPCS) and chaos based spread spectrum image steganography (CSSIS).

Research in hiding data inside image using steganography technique has been done by many researchers, for example in [6-10]. Warkentin *et al*. [6] proposed an approach to hide data inside the audiovisual files. In their steganography algorithm, to hide data, the secret content has to be hidden in a cover message. El-Emam [7], on the other hand, proposed a steganography algorithm to hide a large amount of data with high security. His steganography algorithm is based on hiding a large amount of data (image, audio, text) file inside a colour bitmap (bmp) image. In his research, the image will be filtered and segmented where bits replacement is used on the appropriate pixels. These pixels are selected randomly rather than sequentially. Chen *et al*. [8] modified a method used in [9] using the side match method. They concentrated on hiding the data in the edge portions of the image. Wu *et al*. [10], on the other hand, used pixel-value differencing by partitioning the original image into non-overlapping blocks of two consecutive pixels.

This research uses a similar concept introduced by El-Emam [7]. A bitmap (bmp) image will be used to hide the data. Data will be embedded inside the image using the pixels. Then the pixels of stego image can then be accessed back in order to retrieve back the hidden data inside the image. Two stages are involved. The first stage is to come up with a new steganography algorithm in order to hide the data inside the image and the second stage is to come up with a decryption algorithm using data retrieving method in order to retrieve the hidden data that is hided within the stego image.

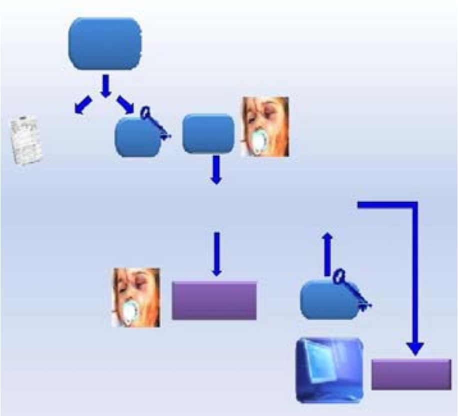
Steganography using 3-dimentional discrete cosine transformation aims on hiding images(s) behind another image. This proposed method transforms a block of integrated composite image i.e, coverimage + targetimage1 + targetimage2 into the frequency domain using 3-dimentional discrete cosine transformation. The proposed algorithm transmits a reddish image (stego-image) from the sender end which is the result of 3 -D DCT and subsequent 2-D IDCT on the composite image (coverimage + targetimage1 + targetimage2). On the receiver end, the individual cover image, targetimage1 and targetimage2 are recovered by using 3-D Inverse discrete cosine transformation and 2-D DCT.

To the best of our knowledge, this concept of 3-D DCT is wholly a new concept of converting a cube of spatial domain values to the frequency domain value. The reverse is done using 3-D IDCT. This concept is based on the concept of 2-D DCT.

Lastly, the main advantage of this proposed method is that two images can be hidden behind one image. Moreover during transmission, a complete new image (different from cover or target images) is transmitted over network thereby reducing the chance of suspicion of the network eavesdropper [9-11].

Proposed Algorithm

Our proposed algorithm is using two layers of security to maintain the privacy, confidentiality and accuracy of the data. Fig. 1 shows the framework for the overall process of the system. The system is able to hide the data inside the image as well as to retrieve the data from the image.



From Fig. 1, for hiding the data, a username and password are required prior to use the system. Once the user has been login into the system, the user can use the information (data) together with the secret key to hide the data inside the chosen image.

For retrieving the data, a secret key is required to retrieving back the data that have been embedded inside the image. Without the secret key, the data cannot be retrieved from the image. This is to ensure the integrity and confidentiality of the data.

In the sender side, one coverimage (C) and two targetimage (T1 and T2) are selected. All the images are resized in the size of coverimage C using bi-cubic interpolation. All the images are converted into gray scale. Then these three images are integrated as a composite image by placing the coverimage (C) in the first plane, targetimage1 (T1) in the next plane ad targetimage2 (T2) in the last plane. Thus this integrated composite image is a color image different from C, T1, T2. Then 3-D DCT is applied on each 4*4*3 cubical segment of the composite image. Of this block obtained by 3-D DCT, the first four values i.e, of the first plane is taken and 2-D IDCT is performed. The other AC components are quantized. Thus the cube is formed where the first plane has large values and the other planes has low values. This is done until the whole composite image is transformed. Then this is transmitted to the receiver end as the stego image which is different from C, T1, T2 and mostly reddish in texture.

In the receiver side a stego image is received. This image is mostly reddish in color. Block of 4*4*3 pixel values is selected. The first four pixels values i.e, of the first plane is taken and 2-D DCT is applied on them. The result is clubbed with the other pixel values of the remaining two layers. The whole cube is then transformed using 3-D IDCT. This is continued until the whole stego image is transformed. The target image T1 is retrieved from the second layer and T2 from the third layer of the transformed stego image.

Sender Side Algorithm

Input: This function will take Target Image1, Target Image2 and Cover Image as input. All the input images must be grayscale image.

Output: It will output the encoded stego-image.

1. Read the Cover Image and 2 target images Target Image1, Target Image2.
2. Resize the 3 images using bi-cubic interpolation to a equal size where both the rows and columns are even.
3. Create two 3-d matrix A of who's each plane is of the size of the resized images and B of size (2X2X3).
4. Store the pixel values of 3 images in the 1st, 2nd and 3rd plane of 'A' respectively.
5. Store the adjacent quadruple values of the 3 planes of A and store it in the corresponding planes of B.
6. Calculate the 3-D DCT of B and store it in B. Then calculate the 2-D IDCT of the 1st plane of B and store it in the 1st plane of B.
7. Store the values of 1st, 2nd and 3rd plane of 'B' in the corresponding quadrant of 1st, 2nd and 3rd plane of 'A' respectively.
8. Move to the next quadrant of 'A' until 'A' is exhausted and repeat step 5 to step 7.
9. Return the new matrix 'A' as Stego-Image.

Receiver Side Algorithm

Input: This function will take Stego-Image as input.
Output: It will output the decoded 2 Images.

1. Read the Stego-Image and calculate its size.
2. Take two 3-d matrix A whose each plane is of the size of Stego-Image and B of size (2X2X3).Store the 1st,2nd and 3rd plane of Stego Image in the corresponding 1st,2nd and 3rd plane of matrix A respectively.
3. Store the quadrant values of the 3 planes of A and store it in the corresponding planes of B.
4. Calculate the 2-D DCT of the 1st plane of B and store it in the 1st plane of B. Then calculate the 3-D IDCT of B and store it in B.
5. Store the values of 1st, 2nd and 3rd plane of 'B' in the corresponding quadrant of 1st, 2nd and 3rd plane of 'A' respectively.
6. Move to the next quadrant of 'A' until 'A' is exhausted and repeat step 3 to step 5.
7. Take 2 matrix C and D of size of the Stego-Image. Store the 2nd and 3rd plane of A in C and D respectively.
8. Retrieve TargetImage1 and TargetImage2 from 2 matrices C and D respectively.

Once the message is hidden inside the image, this message can be extracted back from the stego image. The following algorithm shows the extraction of the secret message from the stego image. In order to retrieve a correct message from the image, a secret key is needed for the purpose of verification. For the data extracting method, a secret key is needed to detect whether the key is match with the key that decodes from the series of binary code. Once the key is matched, the process continues by forming the binary code to a zipped text file, unzip the text file and transfer the secret message from the text file to retrieve the original secret message.

Algorithm

```
Begin
Input: Stego_Image, Secret_Key; Compare Secret_Key;
Calculate BitsPerUnit;
Decode All_Binary_Codes; Shift by 2 unit for bitsPerUnit;
Convert Binary_Codes to Text_File;
Unzip Text_File;
Output Secret_Message;

End
```

The main focuses of this proposed steganography algorithm are the use of transferring secret message to a text file, zipping file, a key, converting both zipped file and key into a series of binary codes, and the use of encoding each last two binary codes into pixels in image. The image quality is still robust where the distortion and colour changes of images are reduced to the minimum or zero-distortion. Secret message, on the other hand, is difficult to be stolen by steganalysis.

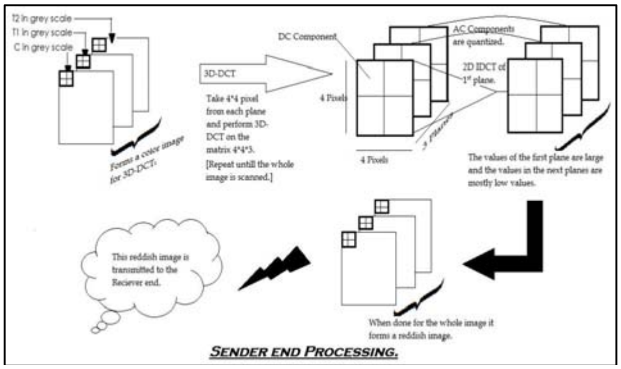

T2 in grey scale
T1 in grey scale
C in grey scale
Forms a color image for 3D-DCT.
3D-DCT
Take 4*4 pixel from each plane and perform 3D-DCT on the matrix 4*4*3.
[Repeat untill the whole image is scanned.]
DC Component
4 Pixels
4 Pixels
3 Planes
AC Components are quantized.
2D IDCT of 1st plane.
The values of the first plane are large and the values in the next planes are mostly low values.
This reddish image is transmitted to the Reciever end.
When done for the whole image it forms a reddish image.
SENDER END PROCESSING.

Fig 2: Sender side approach

This figure shows the sequence of proceedings that are to be performed at the sender end.

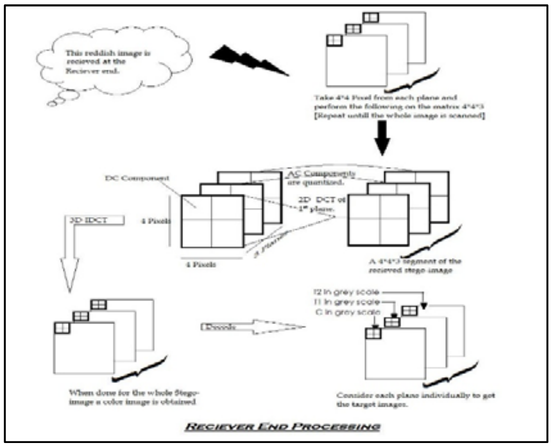

This reddish image is recieved at the Reciever end.
Take 4*4 Pixel from each plane and perform the following on the matrix 4*4*3 [Repeat untill the whole image is scanned]
DC Component
AC Components are quantized.
2D DCT of 1st plane.
3D IDCT
4 Pixels
4 Pixels
3 Planes
A 4*4*3 segment of the recieved stego image
I2 in grey scale
I1 in grey scale
C in grey scale
Decode
When done for the whole Stego-image a color image is obtained
Consider each plane individually to get the target images.
RECIEVER END PROCESSING

Fig 3: Receiver side approach

This figure above shows the sequence of proceedings that are to be performed at the receiver end.

The proposed steganography algorithm consists of two image embedding techniques which are data hiding method and data retrieving method. Data hiding method is used to hide the secret message and the key in cover image while data retrieving method is used to retrieve the key and the hidden secret message from stego image. Hence, data or in particular a secret message, is protected in image without revealing to unauthorized party.

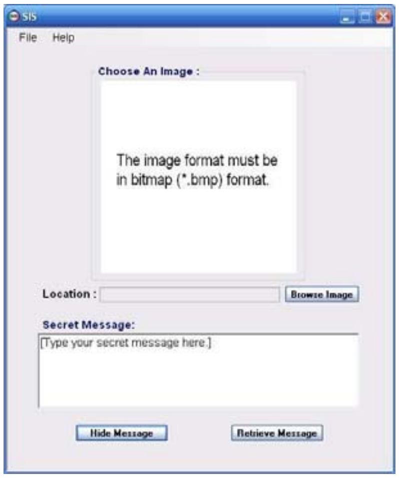
Both from Figs. 2-3 show that 2 layers of security are maintain within the system. However, the secret key is used for verification process in order to retrieve the correct message back from the image. This secret key is also embedded together with the data inside the image. Therefore, when a user is transmitting the image via the internet, that image contains the data and the secret key as well. However, the data can only be retrieved from the image using the system.

Result and Discussion

Based on the proposed algorithm, we develop a simple system, which implements the algorithm. We name the system as Steganography Imaging System (SIS). Based on the framework for the system as seen in Fig. 1, SIS imposed on 2 layers of security. The first layer is for the login purpose and the second layer is for the hiding and retrieving purposes. The system is introduced in [11]. Figs.3 and 4 shows the main interface for the system.

From Fig. 4, SIS has two main boxes, one box for the image and another box for the data that the user needs to hide inside the image. The image box is used for getting the image from any location and the text box is used for hiding and retrieving the message to and from image respectively. In order to hide the data inside the image, a secret key is required for the purpose of security reason. Fig. 5 shows the interface for the secret key which needs to be in 6 characters.

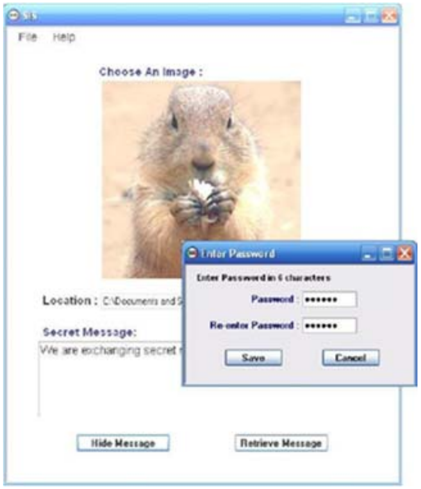
From Fig. 5, the secret key is required to enter twice for the verification purposes. For simplicity, 6 characters are used for the secret key. This secret key is also embedded inside the image together with the data. Therefore, to reduce the size of storing the secret key inside the image, only 6 characters are used for the secret key. Once the data has been key in and the secret key has been entered, the new stego image can be saved to a different image file.



The image format must be in bitmap (*.bmp) format.

[Type your secret message here.]

Fig 4: The main interface for SIS.



Enter Password in 6 characters

Password : ••••••

Re-enter Password : ••••••

Location : C:\Documents and S

We are exchanging secret

Fig 5: The secret key is required for SIS.

This new stego image can then be used by user to send it via internet or email to other parties without revealing the secret data inside the image. If the other parties want to reveal the secret data hidden inside the image, the new stego image file can then be upload again using the system to retrieve the data that have been locked inside the image using the secret key.

From Fig. 6, it shows that the comparison of distortion by naked eyes between cover image and stego image is almost zero. The surfaces of between both images show no difference by using naked eyes even though the size of stego image has a slightly higher than the cover image.

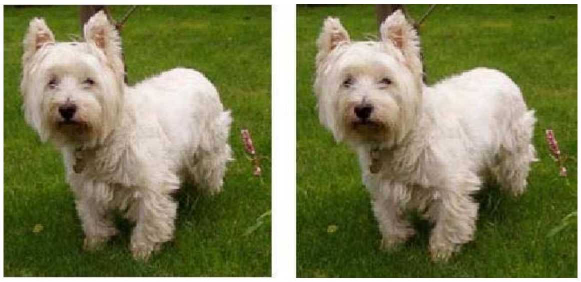


Fig 6: (a) Original image (b) Stego image.

The values show that the stego images have quality images without compromising of the original image.

The pixels of the cover image must fulfill the minimum requirement for the process of data hiding. The minimum image pixel for width is at least 150 while the minimum image pixel for height is at least 112.

Smaller images file size, for example, a BMP image. We then tested the algorithm using the PSNR (Peak signal-to-noise ratio). PSNR is a standard measurement used in steganography technique in order to test the quality of the stego images. The higher the value of PSNR, the more quality the stego image will have.

If the cover image is *C* of size $M \times M$ and the stego image is *S* of size $N \times N$, then each cover image *C* and stego image *S* will have pixel value *(x, y)* from *0* to *M-1* and *0* to *N-1* respectively. The PSNR is then calculated as follows:

$$\text{PSNR}= 10 \log_{10} (\text{MAX}^2/\text{MSE})$$

$$\text{Where MSE}= \sum_{X=0}^{M-1} \sum_{y=0}^{N-1}(c(x, y) - S (x, y))^2$$

Note that MAX is the maximum possible pixel value of the images. For example, if the pixels are represented using 8 bits per sample, then the MAX value is 255.

If the stego image has a higher PSNR value, then the stego image has more quality image. Table 1 shows the PSNR value for two stego images in Figures 6 and 7. The PSNR is calculated using the equation of PSNR in Eq. (1).

Based on values of PSNR from Table 1, the PSNR with a sized of 1.0 MB, is proved to be capable of hiding the Secret Message within it. The biggest size of a zipped file to be encoded into a 1.0 MB BMP image by proposed system is 3.16 KB, which means that the size of image can encodes 10553 characters with spaces (or 1508 words or equally to 4 pages of words) underneath the image with near-zero distortion. Both cover and stego images are alike with the images that showed in Fig. 7 with near-zero distortion noticeable by naked eyes.

Therefore, the proposed steganography algorithm is a strong yet robust algorithm to produce a stego image which will not be doubted by outsider that the image contains any secret message.

The image file format used in proposed algorithm is focused on bitmap (BMP) format. The BMP file format handles graphics files within the Microsoft Windows OS. Typically, BMP files are uncompressed, hence they are large. The advantage of using BMP files is the simplicity and wide acceptance of BMP files in Windows programs. Thus, this type of image is chosen to be used in our proposed algorithm. Since BMP image has a relatively larger size, the pixels in image are relatively larger as well. Thus, it provides more space for binary codes to be encoded within it. To increase as much as characters that can be hidden, zip technique is used to reduce to total size of file and to enhance the security of the file.

Using the proposed algorithm, we test several sizes of BMP images to see the various sizes of data being stored in the image. Table 2 shows these various results for the testing.

Table 1 and 2 shows the comparison of different sizes in BMP image by using the proposed steganography algorithm. These BMP images are used as cover images.

Table 1: The PSNR value of stego images

Image	Reference	PSNR for 1.0 KB embedded
		inside the image
Injured Baby	Figure 7 (a) :	
	Stego Image (1)	76.15
Dog	Figure 7 (b) :	
	Stego Image (2)	81.47

Table 2: Comparison of different sizes in bitmap images

	File Size			Hide	Retrieve
Cover	**Text**	**Zipped**	**Stego**	**Message Message**	
Image	**File**	**File**	**Image**		
		513			
438 KB	4.01 KB	Bytes	584 KB	√	√
438 KB	12.1 KB	4.34 KB	Failed	—	—
1.0 MB	10.4 KB	3.16 KB	1.34 MB	√	√
1.0 MB	10.5 KB	3.15 KB	Failed	—	—
3.14 MB 12.1 KB		4.34 KB	4.19 MB	√	√
3.14 MB 27.0 KB		6.95 KB	4.19 MB	√	√
3.14 MB 54.1 KB		7.03 KB	Failed	—	—
6.74 MB 54.1 KB		7.03 KB	8.99 MB	√	√
9.9 MB	334 KB	8.48 KB	13.2 MB	√	√
9.9 MB	335 KB	8.49 KB	Failed	—	—

to encode the zipped file within it. An image is normally contains 3.14 MB. Using the proposed algorithm, the biggest size of a zipped file that can be hidden into and retrieved from a 3.14 MB BMP image is 6.93 KB, which means that the size of image can encodes 27287 characters with spaces (or 4478 words or equally to 10 pages of words) underneath the image with near-zero distortion.

Conclusions

This paper proposed a new steganography algorithm with 2 layers of security. A system named SIS (Steganography Imaging System) has been developed using the proposed algorithm. We tested few images with various sizes of data to be hidden. With the proposed algorithm, we found that the stego image does not have a noticeable distortion on it (as seen by the naked eyes). We also tested our stego images using PSNR value. Based on the PSNR value of each images, the stego image has a higher PSNR value. Hence this new steganography algorithm is very efficient to hide the data inside the image.

SIS can be used by various users who want to hide the data inside the image without revealing the data to other parties. SIS maintains privacy, confidentiality and accuracy of the data.

References

1. Chen M, Memon N, Wong EK. Data hiding in document images, in: H. Nemati (Ed.). Premier Reference Source–Information Security and Ethics: Concepts, Methodologies, Tools and Applications, New York: Information Science Reference, 2008, 438-450.
2. Lou DC, Liu JL, Tso HK. Evolution of information – hiding technology, in H. Nemati (Ed.), Premier Reference Source–Information Security and Ethics: Concepts, Methodologies, Tools and Applications, New York: Information Science Reference, 2008, 438-450.
3. Schneider, Secrets & Lies, Indiana: Wiley Publishing, 2000.
4. Cole E. Hiding in Plain Sight: Steganography and the Art of Covert Communication, Indianapolis: Wiley Publishing, 2003.
5. Jahnke T, Seitz J. An introduction in digital watermarking applications, principles and problems, in: H. Nemati (Ed), Premier Reference Source–Information Security and Ethics: Concepts, Methodologies, Tools and Applications, New York: Information Science Reference, 2008, 554-569.
6. Warkentin M, Schmidt MB, Bekkering E. Steganography and steganalysis, Premier reference Source–Intellectual Property Protection for Multimedia Information technology, Chapter XIX, 2008, 374-380.
7. El-Emam NN. Hiding a large amount of data with high security using steganography algorithm, Journal of Computer Science 2007; 3:223-232.
8. Chen PY, Wu WE. A modifed side match scheme for image steganography, International Journal of Applied Science & Engineering. 2009; 7:53-60.
9. Chang CC, Tseng HW. A steganographic method for digital image using side match, Pattern Recognition Letters 2004; 25:1431-1437.
10. Wu PC, Tsai WH. A steganographic method for images by pixel-value differencing, Pattern Recognition Letters 2003; 24:1613-1626.
11. Ibrahim R, Kuan TS. Steganography imaging system (SIS): hiding secret message inside an image, Lecture Notes in Engineering and Computer Science: Proceedings of The World Congress on Engineering and Computer Science 2010, San Francisco, USA, 2010, 144-148.