A Novel Approach of Data Hiding in Multimedia Files using Neural Network

Komal Tahiliani* and Dr.N.K.Tiwari**
*Research Scholar
komaltahiliani@yahoo.com
**Bansal Institute of Science and Technology, Bhopal

Abstract: Data hiding is a method of hiding secret messages into a cover-media such that an unintended observer will not be aware of the existence of the hidden messages. We propose new information hiding technique in text, audio, image and video data without embedding any information into the target content by using neural network trained on frequency domain. Proposed method can detect a hidden bit codes from the content by processing the selected feature sub blocks into the trained neural network. Hidden codes are retrieved from the neural network only with the proper extraction key provided. The extraction key, in proposed method, are the coordinates of the selected feature sub blocks and the network weights generated by supervised learning of neural network. The supervised learning uses the coefficients of the selected feature sub blocks as set of input values and the hidden bit patterns are used as teacher signal values of neural network. With our proposed method, we are able to introduce an information hiding scheme with no damage to the target content.

Keywords: Data Hiding, Neural Network.

Introduction

Data hiding [3, 4, 17, 11] is a method of hiding secret messages into a cover-media such that an unintended observer will not be aware of the existence of the hidden messages. In this paper, 8-bit grayscale images are selected as the cover media. These images are called cover-images. Cover-images with the secret messages embedded in them are called stego-images. For data hiding methods, the image quality refers to the quality of the stego-images. The paper is mainly designed for providing security for the data. In this, the sender encrypts the data to some form. While encrypting the data in to some form, the key file is entered by the sender. The purpose of the key file is to provide security to the system as it is known only to the sender and the receiver. Since the actual processing of the data takes place on the remote client the data has to be transported over the network, which requires a secured format of the transfer method. Present day transactions are considered to be "un-trusted" in terms of security, i.e., they are relatively easy to be hacked. And also we have to consider the transfer the large amount of data through the network will give errors while transferring. Nevertheless, sensitive data transfer is to be carried out even if there is lack of an alternative. Network security in the existing system is the motivation factor for a new system with higher-level security standards for the information exchange [1]. This paper proposes the following features. It provides flexibility to the user to secure the data very easily. In this system the data is also hided inside the JPEG Image and Video file. The user who received the file will do the operations like de-embedding, and decryption in their level of hierarchy. People for long time have tried to sort out the problems faced in the general digital communication system but as these problems exist even now, a secured and easy transfer system evolved and came to be known as the Encryption and Decryption of the data and converting the file to JPEG image and video format[30,31] to be transferred using the cryptographic standards The advantages are:

- High level Security
- Cost effective transfer

In this fast growing world where every individual free to access the information on the network and even the people are technically sound enough in hacking the information from the network for various reasons. The organizations have the process of information transfer in and out of their network at various levels, which need the process to be in a secured format for the organizational benefits. The JPEG and Video file that the employee sends reaches the destinations within no time in a JPEG and Video file format where the end user need to de embed the file, decrypt it and use for the purpose.

Features and applications

Data-hiding techniques[38,39,40] should be capable of embedding data in a host signal with the following restrictions and features:

1. The host signal should be non objectionably degraded and the embedded data should be minimally perceptible. (The goal is for the data to remain *hidden*. As any magician will tell you, it is possible for something to be hidden while it remains in

plain sight; you merely keep the person from looking at it. We will use the words *hidden*, *inaudible*, *unperceivable*, and *invisible* to mean that an observer does not notice the presence of the data, even if they are perceptible.)

2. The embedded data should be directly encoded into the media, rather than into a header or wrapper, so that the data remain intact across varying data file formats.

3. The embedded data should be immune to modifications ranging from intentional and intelligent attempts at removal to anticipated manipulations, e.g., channel noise, filtering, resembling, cropping, encoding, lossy compressing, printing and scanning, digital-to-analog (D/A) conversion, and analog- to-digital (A/D) conversion, etc.

4. Asymmetrical coding of the embedded data is desirable, since the purpose of data hiding is to keep the data in the host signal, but no t necessarily to make the data difficult to access.

5. Error correction coding1 should be used to ensure data integrity. It is inevitable that there will be some degradation to the embedded data when the host signal is modified.

6. The embedded data should be self-clocking or arbitrarily re-entrant. This ensures that the embedded data can be recovered when only fragments of the host signal are available, e.g., if a sound bite is extracted from an interview, data embedded in the audio segment can be recovered. This feature also facilitates automatic decoding of the hidden data, since there is no need to refer to the original host signal.

Applications

Trade-offs exists between the quantity of embedded data and the degree of immunity to host signal modification. By constraining the degree of host signal degradation, a data-hiding method can operate with either high embedded data rate, or high resistance to modification, but not both. As one increases, the other must decrease. While this can be shown mathematically for some data-hiding systems such as a spread spectrum, it seems to hold true for all data-hiding systems. In any system, you can trade bandwidth for robustness by exploiting redundancy. The quantity of embedded data and the degree of host signal modification vary from application to application. Consequently, different techniques are employed for different applications. Several prospective applications of data hiding .An application that requires a minimal amount of embedded data is the placement of digital water mark. The embedded data are used to place an indication of ownership in the host signal, serving the same purpose as an author’s signature or a company logo. A second application for data hiding is tamper-proofing. It is used to indicate that the host signal has been modified from its authored state. Modification to the embedded data indicates that the host signal has been changed in some way. A third application, feature location, requires more data to be embedded. In this application, the embedded data are hidden in specific locations within an image. It enables one to identify individual content features, e.g., the name of the person on the left versus the right side of an image. Typically, feature location data are not subject to intentional removal. However, it is expected that the host signal might be subjected to a certain degree of modification, e.g., images are routinely modified by scaling, cropping, and tone scale enhancement. As a result, feature location data hiding techniques must be immune to geometrical and no geometrical modifications of a host signal.

Data Hiding Various Streams

Data hiding in still images

Data hiding in still images [26] presents a variety of challenges that arise due to the way the human visual system (HVS) works and the typical modifications that images undergo. Additionally, still images provide a relatively small host signal in which to hide data. A fairly typical 8-bit picture of 200 □□200 pixels provides approximately 40 kilobytes (kb) of data space in which to work. This is equivalent to only around 5 seconds of telephone-quality audio or less than a single frame of NTSC television. Also, it is reasonable to expect that still images will be subject to operations ranging from simple affine transforms to nonlinear transforms such as cropping, blurring, filtering, and lossy compression. Practical data-hiding techniques need to be resistant to as many of these transformations as possible. Despite these challenges, still images are likely candidates for data hiding. There are many attributes of the HVS that are potential candidates for exploitation in a data-hiding system, including our varying sensitivity to contrast as a function of spatial frequency and the masking effect of edges (both in luminance and

The HVS has low sensitivity to small changes in luminance, being able to perceive changes of no less than one part in 30 for random patterns. However, in uniform regions of an image, the HVS is more sensitive to the change of the luminance, approximately one part in 240. A typical CRT (cathode ray tube) display or printer has a limited dynamic range. In an image representation of one part in 256, e.g., 8-bit gray levels, there is potentially room to hide data as pseudorandom changes to picture brightness. Another HVS “hole” is our relative insensitivity to very low spatial frequencies such as continuous changes in brightness across an image, i.e., vignetting.

An additional advantage of working with still images is that they are non causal. Data-hiding techniques can have access to any pixel or block of pixels at random.

Using these observations, we have developed a variety of techniques for placing data in still images. Some techniques are more suited to dealing with small amounts of data, while others to large amounts. Some techniques are highly resistant to geometric modifications, while others are more resistant to non geometric modifications, e.g., filtering. We present methods that explore both of these areas, as well as their combination.

Data hiding in audio

Data hiding [36, 37] in audio signals is especially challenging, because the human auditory system (HAS) operates over a wide dynamic range. The HAS perceives over a range of power greater than one billion to one and a range of frequencies greater than one thousand to one, Sensitivity to additive random noise is also acute. The perturbations in a sound file can be detected as low as one part in ten million (80 dB below ambient level). However, there are some “holes” available. While the HAS has a large dynamic range, it has a fairly small differential range. As a result, loud sounds tend to mask out quiet sounds. Additionally, the HAS is unable to perceive absolute phase, only relative phase. Finally, there are some environmental distortions so common as to be ignored by the listener in most cases. We exploit many of these traits in the methods.

Data hiding in text

Soft-copy text is in many ways the most difficult place to hide data [34, 35]. (Hard-copy text can be treated as a highly structured image and is readily amenable to a variety of techniques such as slight variations in letter forms, kerning, baseline, etc.) This is due largely to the relative lack of redundant information in a text file as compared with a picture or a sound bite. While it is often possible to make imperceptible modifications to a picture, even an extra letter or period in text may be noticed by a casual reader. Data

Hiding[7,8,9,10] in text is an exercise in the discovery of modifications that are not noticed by readers. We considered three major methods of encoding data: open space methods that encode through manipulation of white space (unused space on the printed page), syntactic methods that utilize punctuation, and semantic methods that encode using manipulation of the words themselves.

Data Hiding in Video

A video data embedding scheme in which the embedded signature data is reconstructed without knowing the original host video [29,24]. The proposed method enables high rate of data embedding [2] and is robust to motion compensated coding, such as MPEG-2. Embedding is based on texture masking and utilizes a multi-dimensional lattice structure for encoding signature information. Signature data is embedded in individual video frames using the block DCT. The embedded frames are then MPEG-2 coded. At the receiver, both the host and signature images are recovered from the embedded bit stream[22,23].

Tradition Method used in Data Hiding

Least Significant Bit (LBS) Method

In computing, the **least significant bit** (**LSB**) [18,19,20,21]is the bit position in a binary integer giving the units value, that is, determining whether the number is even or odd. The lsb is sometimes referred to as the right-most bit, due to the convention in positional notation of writing less significant digit further to the right. It is analogous to the least significant digit of a decimal integer, which is the digit in the ones (right-most) position.

In referencing specific bits within a binary number, it is common to assign each bit a bit number, ranging from zero upwards to one less than the number of bits in the number. However, the order used for this assignment may be in either direction. Both orderings are used (in different contexts), which is why "lsb" is often used to designate the units bit instead of a bit number, which has the potential for confusion.

By extension, the least significant bits (plural) are the bits of the number closest to, and including, the lsb.The least significant bits have the useful property of changing rapidly if the number changes even slightly. For example, if 1 (binary 00000001) is added to 3 (binary 00000011), the result will be 4 (binary 00000100) and three of the least significant bits will change (011 to 100). By contrast, the three most significant bits stay unchanged (000 to 000).

Neural Networks

In information technology [1,5,6] a neural network is a system of programs and data structures that approximates the operation of the human brain. A neural network [41,33,28,27,25] usually involves a large number of processors operating in parallel, each with its own small sphere of knowledge and access to data in its local memory. General Regression Neural Network (GRNN), a simple, one-parameter neural networks model, is proposed for the Embed and De Embed process. Neural networks a very powerful and general framework for representing non-linear mapping from several input variables to several output variables. The process to determining the values of these parameters on the basis of a data set is referred to as learning or training, and so the data set is generally referred to as a training set. A neural network can be viewed as suitable choice for the functional forms used for Embed and De Embed processes.

Proposed Method

Multilayer Perceptron Algorithm

We take use of Multilayer Perceptron network[32] to train and simulate images. This BP neural network uses three levels: input level, hidden level and output level. In neural network, the important issue is the slow of convergence. In practice, this is the main limitation of neural network applications. And many new algorithms claimed fast convergence were developed. In this paper a single parameter dynamic search algorithm is used to accelerate network train. Each time only one parameter to be searched to achieve best performance, so this learning algorithm has a better improvement than other old algorithms [2,3]. We set the number of this network's input as features, and node number of hidden level is set to be 40, and output is either yes or no.

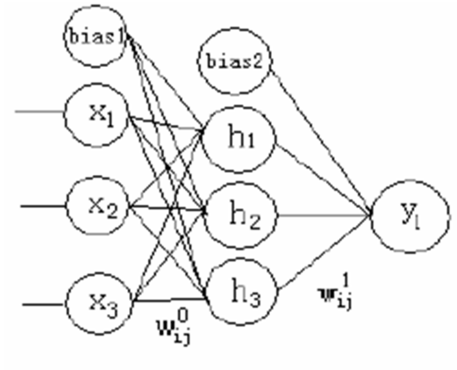

Fig.1. Three level BP neural network

Normally sigmoid function is used for this model and is expressed as follows

$$f(x) = \frac{1}{1+e^{-x}}$$

Each synaptic link has a network weight. The network weight from unit *i* to unit *j* is expressed as *wij* and the output value for unit *i* is expressed as *Oi*. The output values for the unit is determined by the network weight and the input signal. Consequently, to change the output value to a desired value, adjustment of these network weights are needed.

In proposed method, we use back propagation learning as learning method. Back propagation learning is a supervised learning. This method tries to lower the difference between the teacher signal and the output signal by changing the network weight. Changes of the network weight according to the difference in the upper layer propagate backward to the lower layer. This difference between the teacher signal values are called as error and often expressed as δ . When teacher signal t_k is given to the unit *k*

of output layer, the error δ_k will be calculated by following function:

$$\delta_k = (t_k - O_k) \cdot f'(O_k)$$

To calculate the error value *δj* for hidden unit, error value *δk* of the output unit is used. The function to calculate the error value *δj* for hidden unit *j* is as follows:

$$\delta_j = (\sum_k \delta_k w_{jk}) \cdot f'(O_j)$$

After calculating the error values for all units in all layers, then network can change its network weight. The network weight is changed by using following function:

$$\Delta w_{ij} = \eta \delta_j O_i$$

η in this function is called learning rate. Learning rate is a constant which normally has a value between 0 and 1 and generally represents the speed of learning process.

Steps of back-propagation algorithm

Initialize the weights-The weights in the network are initialized to small random numbers (that is ranging from –1.0 to 1.0 or –0.5 to 0.5). Each unit has a bias associated with it. Propagate the input forward- In this step the net input and output of each unit in the hidden and output layers are computed. Given a unit j in a hidden or output layer, the net input, Ij to unit j is

$Ij = \sum i \quad wijOi + \theta j$

Where wij is the weight of the connection from i in the previous layer to unit j; Oi is the output of unit i from previous layer; and θj is the bias of the unit.

The output of unit j, is computed as

$Oj = 1/ (1+ e –Ij)$

Back propagate the error: -The error is propagate backward by updating the weights and biases to reflect the error of the network’s prediction. For a unit j in the output layer, the error Errj is computed by where Tj is the true output.

$Errj = Oj (1-Oj)(Tj-Oj)$

To compute the error of a hidden layer unit j, the weighted sum of the errors of the units connected to unit j in the next layer are considered. The error of a hidden layer unit j is

$Errj = Oj (1-Oj) \sum k \; Errk \; wjk$

Where wjk is the weight of the connection from unit j to a unit k in the next higher, and Errk is the error of unit k.
Now the weight and biases are update to reflect the propagate errors.

$\Delta \; wij = (l) \; Errj \; Oj$
$wij = wij + \Delta \; wij$
Where l is the learning rate.

$\Delta\theta j = l \; Errj$
$\theta j = \theta j + \Delta\theta j$

Terminating Condition- Training stop when all Δ wij in the previous epoch were so small as to be below some specified threshold, or the percentage of samples misclassified in the previous epoch is below some threshold or a pre specified number of epochs has expired.

Information hiding algorithm

For embedding procedure, we must decide the structure of neural network.[27] The amount of units for input layer is decided by the number of pixels selected as feature values from the content data. We select unique feature sub blocks in obedience to the teacher signal. Proposed method uses the diagonal values of this selected feature sub block as input values for the neural network. For better approximation, one bias neuron is added for input layer. The neural network is trained to output a value of 1 or 0 as an output signal. In proposed method, one network represents one binary digit for corresponding secret codes. The adequate amount of neurons in the hidden layer[14], for back propagation learning in general, is not known. So the number of neurons in hidden layer will be taken at will. In proposed method, ten hidden units are used. For better approximation, one bias neuron like in the case for input layer is introduced for the hidden layer. After the learning process, the network weights are converged to certain values. We use this network weights and the coordinates of selected feature sub blocks are saved as extraction keys.For extraction process, same neural network structure is constructed. This can be constructed by having the proper network weights. Only with the proper input values of the selected feature sub blocks[15] will output the corresponding hidden codes. And proper input values are induced only when you know the proper coordinates of the sub blocks for the corresponding hidden signal. This relationship is shown in Figure 2 and 3.

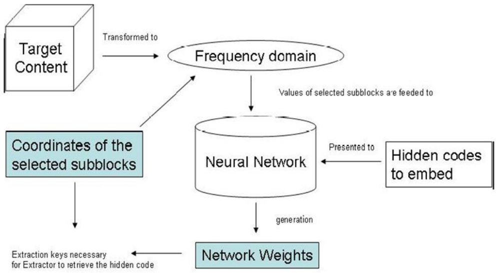
Target Content
Transformed to
Frequency domain
Values of selected subblocks a'e feeded to
Coordinates of the selected subblocks
Neural Network
Presented to
Hidden codes to embed
generation
Extraction keys necessary
for Extractor to retrieve the hidden code
Network Weights

Figure 2. Embedding procedure

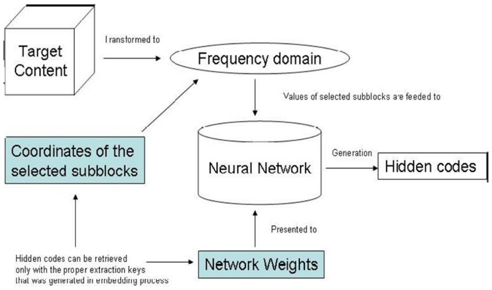
Target Content
I ransformed to
Frequency domain
Values of selected subblocks a'e feeded to
Coordinates of the selected subblocks
Neural Network
Generation
Hidden codes
Presented to
Hidden codes can be retrieved
only with the proper extraction keys
that was generated in embedding process
Network Weights

Figure 3. Extracting procedure

Conclusion

In this paper, we proposed an information hiding technique without embedding[12,13,14] any data into target content. This characteristic is effective when user must not damage the content but must conceal a secret code into target content. Proposed method uses multi-layered neural network model for classifying the input patterns to corresponding hidden signals.

Future Enhancements

Any specification untraced errors will be concentrated in the coming versions, which are planned to be developed in near future. The following have been identified as the future scope:

1. Implementation of the cryptographic algorithm in hardware.
2. Error correction using various error correction techniques or development of new techniques.
3. Data compression using existing techniques or developing of new techniques
4. This package can be used across WAN and MAN Networks with further implementations for the existing code.
5. This package can be used in different sectors such as organizations, Indian Defense and Institutions etc.

References

[1] H. Sasaki, editor. Intellectual Property Protection for Multimedia Information Technology. IGI Global, 12 2007.

[2] D. Kahn. The history of steganography. In Proceedings of the First International Workshop on Information Hiding, pages1.5, London, UK, 1996. Springer-Verlag.

[3] S. Katzenbeisser and A. P. Fabien, editors. Information Hiding Techniques for Steganography and Digital Watermarking. Artech House Publishers, 1 2000.

[4] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker. Digital Watermarking and Steganography. Morgan Kaufmann, 2 edition, 11 2007.

[5] I. Cox, J. Kilian, T. Leighton, and T. Shamoon. Secure spread spectrum watermarking for images, audio and video. Image Processing, 1996. Proceedings., International Conference on, 3, 1996.

[6] F. Rosenblatt. The perceptron a probabilistic model for information storage and organization. Brain Psych. Revue,62:386.408, 1958.

[7] D. Rumelhart and J. McClelland. Parallel distributed processing: explorations in the microstructure of cognition, vol.1: foundations. MIT Press Cambridge, MA, USA, 1986.

[8] Bender, W., Morimoto, N. and Lu, A., "Techniques for data hiding", *IBM Syst. J.,* vol. 35, no. 3/4, pp. 313–336, 1996.

[9] Chan, C.K. and Cheng, L.M., "Hiding data in images by simple LSB substitution", *Pattern Recognition*, vol. 37, no. 3, pp. 469–474, 2004.

[10] Chang, C.C., Hsiaob, J.Y. and Chan, C.S., "Finding optimal least-significant-bit substitution in image hiding by dynamic programming strategy", *Pattern Recognition*, vol. 36, pp. 1583 – 1595, 2003.

[11] Chang, C.C. and Tseng, H.W., " Data hiding in images by hybrid LSB substitution ", *3rd International Conference on Multimedia and Ubiquitous Engineering*, art. no. 5318917, pp. 360-363, 2009.

[12] Chen, T.S., Chang, C.C. and Hwang, M.S., "A virtual image cryptosystem based upon vector quantization", *IEEE Trans. Image Process*. vol. 7, no. 10, pp. 1485–1488, 1998.

[13] Chung, K.L., Shen, C.H. and Chang, L.C., "A novel SVD- and VQ-based image hiding scheme", *Pattern Recognition Lett*., vol. 22, no. 9, pp. 1051–1058, 2001.

[14] Johnson, N.F. and Jajodia, S., "Exploring Steganography: Seeing the Unseen", *IEEE Computer Journal*, vol. 31, no.2, pp. 26-34, 1998.

[15] Katzenbeisser, S. and Petitcolas, F.A.P., *Information Hiding Techniques fo Steganography and Digital Watermarking,* Artech house, Inc., 2000.

[16] Krutz, R.D., Consulting Editor, *Hiding in plain sight: Steganography and the Art of Cover communication*, Wiley Publishing, Inc., 2003.

[17] Li, X. and Wang, J., "A steganographic method based upon JPEG and particle swarm optimization algorithm", *Information Sciences*, vol. 177, no. 15, pp. 3099–3109, 2007.

[18] Marvel, L.M., Boncelet, C.G. and Retter, C.T. (1999), "Spread spectrum image steganography", *IEEE Trans. Image Process*., vol. 8, no. 8, pp. 1075–1083.

[19] Schneier, B., *Applied Cryptography Second Edition: Protocols, Algorithms, and Source Code in C,* Wiley Computer Publishing, John Wiley & Sons, Inc., 1996.

[20] Wang, R.Z., Lin, C.F. and Lin, J.C., "Hiding data in images by optimal moderately significant-bit replacement", *IEE Electron. Lett*., vol. 36, no. 25, pp. 2069–2070, 2000

[21] Wang, R.Z., Lin, C.F. and Lin, J.C., "Image hiding by optimal LSB substitution and genetic algorithm", *Pattern Recognition*, vol. 34, no. 3, pp. 671–683, 2001.

[22] Jack Kelley. Terror groups hide behind Webencryption. USA Today, Feburary 2001. http://www.usatoday.com/life/cyber/tech/2001-02-05-b inladen.htm.

[23] Neil F.Johnson and Sushhil Jajodia. Steganalysis: The Investigation of Hidden Information. Proceedings of the IEEE Information Technology Conference, Syracuse, New York,USA.1998

[24] N. F. Johnson and S. Jajodia, “Steganalysis of Images Created Using Current Steganography Software,” Lecture Notes in Computer Science, vol.1525, Springer-Verlag, Berlin, 273-289, 1998

[25] H. Farid, “Detecting Steganographic Message in Digital Images”, Technical Report, TR2001-412, Dartmouth College, Computer Science, 2001

[26] I.Aveibas, N.Memon, B.Sankur. Steganalysis based on image quality metrics. Multimedia Signal Processing, 2001 IEEE Fourth Workshop on , 2001 Page(s): 517 -522

[27] R. Chandramouli and N. Memon, "Analysis of LSB based Image Steganography Techniques", Proceedings of ICIP 2001, Thessaloniki, Greece, 2001.

[28] J. Fridrich, R. Du, and L. Meng, “Steganalysis of LSB Encoding in Color Images,” Proceedings IEEE International Conference on Multimedia and Expo, New York ,2000

[29] J. Fridrich, M. Goljan, and R. Du, "Steganalysis based on JPEG compatibility," SPIE Multimedia Systems and Applications IV, August 20−24, 2001.

[30] N. F. Johnson and S. Jajodia, “Steganography: Seeing the Unseen,” *IEEE Computer*, February, 26−34,1998

[31] [10] Jessica Fridrich, Miroslav Goljan. Practical Steganalysis of Digital Images – State of the Art. Preprint 2001

[32] Xuefeng Wang and Yingjun Feng. A New Learning Algorithm for Neural Networks. Journal of Harbin institute of technology. 29(2):23-25

[33] Xuefeng Wang and Yingjun Feng. A Fast Learning Algorithm of Multi-Layer Neural Network. OR Transactions. 2(3):25-29 1998

[34] B. Chen and G. W. Wornell. Implementations of quantization index modulation methods for digital watermarking and information embedding of multimedia. Special Issue on Multimedia Signal Processing, vol. 27:7-33, 2001

[35] Voloshynovskiy, S.,Herrigel and Rytsar Y. StegoWall: Blind statistical detection of hidden data. Proceedings of SPIE. Vol. 4675:57-68.2002.

[36] R. Z. Wang, C. F. Lin and J. C. Lin, Hiding data in images by optimal moderately significant-bit replacement, IEE Electronics Letter, 36,(2000) 2069-2070.

[37] C. K. Chan and L. M. Cheng, Improved hiding data in images by optimal moderately significant-bit replacement, IEE Electronics Letter, 37,(2001)1017- 1018.

[38] R. Z.Wang, C. F. Lin and J. C. Lin, Image hiding by optimal LSB substitution and genetic algorithm, Pattern Recognition Letter, 34,(2001) 671-683.

[39] M. S. Hwang, C. C. Chang and K. F. Hwang, Digital watermarking of images using neural networks, Journal of Electronic Imgaging, 9, (2000) 548-555.

[40] P. T. Yu, H. H. Tsai and J.S. Lin, Digital watermarking based on neural networks for color images, Signal processing, 81,(2001)663-671.

[41] C.C. Chang and I. C. Lin, Robust image watermarking system using neural network, Intelligent Watermarking Techniques (World Scientific, Singapore 2004) 395-427.