Network and Communication Technologies; Vol. 4, No. 1; 2019
ISSN 1927-064X E-ISSN 1927-0658
Published by Canadian Center of Science and Education

Secure Image Steganography Algorithm Using Radial Basis Function Neural Network

Areej Abed Hutaibat[1]

[1] Social Security Corporation, Amman, Jordan

Correspondence: Areej Abed Hutaibat, Social Security Corporation, Amman, Jordan. E-mail: hutabat84@yahoo.com

Received: March 20, 2019 Accepted: May 5, 2019 Online Published: May 10, 2019

doi:10.5539/nct.v4n1p16 URL: https://doi.org/10.5539/nct.v4n1p16

Abstract

Recently, ensuring the security of secret messages over computer networks has significantly increased in importance. For this reason, a new system is proposed that tries to hide text using Artificial Neural Network (ANN), and more precisely using Radial Based Function, with zero mean square error, in addition to encryption techniques, to make sure that the resulting text is exactly the same as the one that was sent.

In this study the text is encrypted by an ordinary encryption algorithm, then the encrypted text will be embedded within the image and the positions of each encrypted text value will be determined, and in the last step the taken values (positions) will be encrypted using the neural network. The resulting encrypted text is unpredictable, making it very secure.

On the receiver side, only the person, who has knowledge of the decryption key, neural network inputs P and parameters, will be able to see the original message embedded in the image.

Keywords: steganography, artificial neural network, radial basis function

1. Introduction

Security and privacy are very important features in a modern system. Any communication process will be useless if it is unsecure. Intruders try to take a copy of the message to steal important information, and they may alter some information or delete it. Many security methods were proposed to protect the transmission lines. Cryptography and steganography are examples of these methods. One of the problems that faces cryptography is that by sending the encrypted text (cipher text) anyone in the middle way between sender and receiver can understand that there is some secret information in this cipher text. Steganography, on the other hand, hides the secret message in the cover media, so nobody in the middle way between sender and receiver can find out that there is an important data hidden in the sending file.

To ensure confidentiality and to amplify the strength, the author aims to implement a new method for hiding secret messages, which combines both cryptography for privacy and steganography for secrecy using neural network. The scope of this paper is achieving high level security by the combination of steganography and cryptography properties in such a way that it is harder for a steganalyst to obtain the secret message (Geetha and Prasad, 2014; Kaur et al., 2014; Siddharth and Siddiqui, 2012).

This paper is organized in five sections. The current section, section one, is the introduction, section two illustrates the techniques used in this study, such as steganography, its history, methods and implementation. Also cryptography is described because this study's method is a combination of steganography and cryptography. Furthermore, it provides details about Radial Basis Function (RBF) neural network, why the adoption of RBF, its characteristics and its structure. The author discusses the proposed method in section three, and how the steganography process is done at both at the sender side and receiver side is described in more details. Section four gives the performance metrics that wereused in order to prove the efficiency of the proposed method, finally in section five this work is concluded.

2. Techniques Background

2.1 Steganography

In cryptography the text is converted to unreadable form, steganography on the other hand hides the existence of

this text, so nobody can detect the presence of it which improves security. Steganography is a combination of two words derived from a Greek words, “stegos” which means cover and “grafia” which means writing. Hence, steganography means covered writing. Different cover media is used to hide the message like images, audio and video.The most common one is images, also the message itself can be text, image and audio (Kaur et al., 2014).

2.1.1 Steganography Methods

- Injection: Injection steganography is a process in which a secret message is embedded in a cover file, so the cover-file size will be increased because the embedded data is added to the original data. Any file type can be used with injection and the output file is called a stego-file.
- Substitution: Substitution steganography is a process in which part of the cover-file is replaced with the secret message. The selection of the replaced part is based on which one is not used or rarely used and silent area in audio file cases, best fit in spaces that are blank and executable files. However, replacing parts in executable files will cause errors.
- Propagation: Propagation steganography is a process in which no cover file is used, instead a software is utilized which will take a part of the secret message, then the output file will be generated. The same output file obtained each time the software is run, is known as “mimic”. Then applying a generation engine on the mimic will generate the original data. [30].

2.1.2 Steganography Implementation

Steganography’s basic elements are: the carrier image which called “cover image”, the resulting image, which has the hidden data called "stego image" and the secret key that is shared between the participating parties controlling the embedding process, with this key only the recipient who knows it will extract the secret message

2.1.3 Steganography Principles

The standard principles that are used to measure the performance of any given steganography method are (Usha et al., 2013):

- Amount of Data: The basic idea behind steganography is to hide information as much as possible within a file.
- Ease of Detection: When hiding information, it is important to be sure that it is very difficult for someone to detect.

2.2 Cryptography

Cryptography is a Greek word meaning “secret writing”. It is a method of transforming data in a way that renders it unreadable by anyone except the intended recipient (Narayana & Prasad, 2010). The need for cryptography emerged during wars, as there was need for a secure method to send military information to their Allies or Armies. If the enemy caught the messenger and the message was not encrypted they could read it, so when a message is sent, it must be written in a way that only the authorized person can read it. Also the messenger must be unable to read the message that he or she carries. In World War I and World War II, cryptography witnessed a significant evolution, but the largest evolution was after the invention of the computer networks. In modern days many ciphering algorithms developed and introduced spreading computer cryptography.

2.2.1 Cryptography Methods

- Substitution Cryptography: Converts one letter/number to another, the sender and receiver in this method must have a map for the letters/numbers to encrypt/decrypt the message (Van Tilborg, 1999).
- Transposition Cryptography: Changes position of letter in text, here the sender and the receiver must know the position of the letter to change it in encryption/decryption process (Van Tilborg, 1999).

2.2.2 Cryptography Importance

The importance of cryptography stems from many aspects that aim to protect the information from changing or being deleted, or seen by unauthorized persons. These aspects are:

Confidentiality – Only the authorized person is allowed to view the message.

Integrity – Ensuring that the message was not altered by any third party or an unauthorized one.

Authenticity – Validation process of the message source, or process to ensure that the sender is properly identified.

Non repudiation – Creation of an identity for the sender, so that no one can deny who has sent the message.

Access Control – Provision of access to an object, which requires access to the associated crypto keys.

2.2.3 Cryptography Process

All encryption/decryption algorithms have main common characteristics or steps to secure the data or information. These elements are shown in Figure 1 (Narayana & Prasad, 2010).

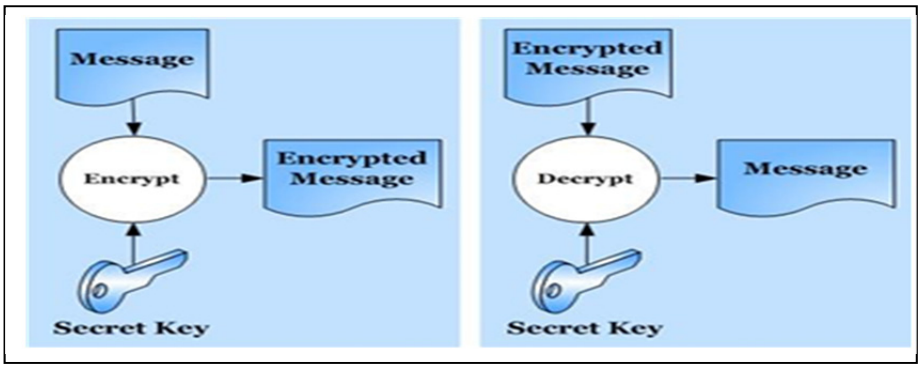

Figure 1. Cryptography Process courtesy of [4]

Message– A message in its natural format readable by an attacker.

Encrypted Message – Message altered to be unreadable by anyone except the intended recipients.

Secret Key – a unique Sequence that used in the cryptographic algorithm for each encryption and decryption.

2.2.4 Encryption Modes

2.2. 4.1 Symmetric Key Algorithm

Both the sender and receiver use the same key to encrypt the Plaintext or to decrypt the Cipher-text. This key is called “Secret”, and it must be sent over a secure channel while the Cipher-text could be sent over any insecure channel.

2.2.4.2 Asymmetric Key Algorithm

Sometimes also called public-key cryptography. This is a type of cryptography where two keys are used to encrypt /decrypt the message. Initially, sender/receiver receives two keys, a public and private key; both keys are sent by a certificate authority, this certificate authority must send each key in a different channel to ensure security.

2.3 Radial Basis Function Networks

A radial basis function (RBF) neural network is a network that operates on a feed-forward basis through supervised training algorithms. The configuration takes place with a single hidden layer of units whose activation function is selected from a set of basis functions. These radial basis functions have various advantages, despite their similarities to back propagation. Firstly, they train much faster than back propagation networks. Secondly, they are less susceptible to problems with non-stationary inputs because of the radial basis function hidden units behavior (Halici, 2001).

RBF networks have a neural network architecture, which supports their usefulness. This usefulness is derived from the main difference between feed forward networks and back propagation networks, which is the single hidden layer behavior. This layer uses a Gaussian or other basis function, instead of sigmoidal activation functions like back progagatio networks. The hidden units in the RBF network function as a processor to compute the match between the input vector and its connection weights. Essentially, these basis units are considered as a good pattern detector. The weights that connect the basis units to the outputs use the hidden units in linear combinations to create the final classification or output.

In this section, first the characteristics of RBF network will be introduced, then the structure of the network will be introduced and an explanation is given of how it can be used for data interpolation and function approximation. Then another explanation is given of how it can be trained, and finally the commonly used

function used in RBF will be introduced.

2.3.1 Characteristics of RBF Network

- They are feed-forward networks with two-layers.
- A set of radial basis functions are implemented of the hidden nodes.
- Linear summation functions is implemented at output nodes as in a Multi-Layer Perceptron (MLP) networks
- The RBF neural network training process needs two phases: first the weights from the input to hidden layer are determined, second these weights are feeded to output layer.
- The learning (training) is very fast.
- The networks are very good at interpolation.

2.3.2 The Structure of the RBF Networks

The RBF neural network is a kind of feed-forward neural network. The RBF neural network comprises of three layers: input layer, output layer and a hidden layer in between which is a layer of processing units known as hidden units. Each of these implements a radial basis

a. Input Unit: The input layer is made up of source whose number is equal to the Dimension p of the input vector u.

b. **Hidden Unit:** The second layer is the hidden layer which is composed of nonlinear units that are connected to all of the input layer nodes directly; each hidden unit receives input from all the nodes at the components at the input layer. The hidden unit incorporates a radial basis function, which has two main parameters: center and width. Vector ci is the center of the basis function for a node i at the hidden layer. Its size is the same as the input vector u and each unit in the network normally has a different center. First, the radial distance di, between the input vector u and the center of the basis function ci is computed for each unit i in the hidden layer as (Halici, 2001).

$$d_i = |u\text{-}c_i| \tag{1}$$

c. Using the Euclidean Distance: Then, the output hi of each hidden unit i is calculated by applying the basis function G to this distance (6)

$$hi = G(d_i, \sigma_i) \tag{2}$$

The basis function is a curve (typically a Gaussian function, the width corresponding to the variance, σi) which has a peak at zero and decreases as the distance increases from the center (Gurney, 1999).

d. Output Layer: The conversion from input unit to the hidden unit space is nonlinear, but the conversion from the hidden unit space to the output unit space is linear. The jth output is computed as (Olanrewaju et al., 2012)

$$x_j = w_{0j} + \sum_{i=1}^{L} w_{ij}\, h_i \quad \text{j=1, 3….M} \tag{3}$$

2.3.3 Training RBF Networks

The training of a RBF network is identified as a nonlinear, unconstrained optimization problem as is given below:

Given, input output training patterns (uk, yk), k=1, 3..K, choose wi, j and ci,, i=1,3...L, j=1, 3...M in order to minimize

$$J\,(w, c) = \sum_{k=1}^{K} |y^k - f(u^k)|^3 \tag{4}$$

Note that if ci's (radial basis function centers values) are known, the training problem becomes quadratic (Halici, 2001).

2.3.4 Adjusting the Widths

At the most basic level, the hidden units in the RBF network have the same width to inputs. However, in parts of the input space with few patterns it is preferred to have hidden units with a wide area of reception. And, in parts of the input space, it might be preferred to have very highly tuned processors with narrow reception fields. The computation of the individual widths increases the performance of the RBF network instead of a more complicated training process.

2.3.5 Adjusting the Centers

In a back propagation network, all weights in all layers are adjusted at the same time. In RBF networks, on the other hand, the weights into the hidden layer basis units are set prior the second layer weights being fine-tuneds. The input moves away from the connection weights, and the activation value drops. This behavior has created the term “center” for the first-layer weights. Using “Kohonen” feature maps these weights can be calculated, but also statistical methods such as K-Means clustering, or some other methods can be used. These are used to set the areas of sensitivity for the RBF network’s hidden unit, which remains fixed.

2.3.6 Adjusting the Weights

Once hidden layer weights are set, a second phase of training leads to the adjustment of output weights. This process uses the standard steepest algorithm. Note that the training task becomes quadratic once if ci’s (RBF centers) are known.

3. The Proposed Method

In this paper, the author develops a new image steganography technique, as the image is consistent with a matrix of pixels. The study will be carried out on that matrix, the image representation used is grayscale, which is adopted for simplicity. The same algorithm is applicable on RGB images, but three matrices have to be managed, one for each color, so we do the coding on the grayscale to deal with one matrix. This process is divided into two main processes, Encryption and Decryption. The encryption process is done by the sender and the decryption is done by the receiver. In this section, the author will briefly discuss the whole process, and will take a quick look at the coding process. Matlab 9.0 software was used to develop this code, which was good choice because it offers a large library for image processing, and learning any command is easy.

3.1 Steganography Process

As stated in the introduction, the steganography process is combined with the cryptography process to improve security. First, the secret message that needs to be transmitted will be encrypted by any encryption algorithm. At the same time the covered media (image) will be converted from multi-dimensional array into a one-dimensional array. Second, a neural network will be created. This neural network is of type RBF and is applied to the encrypted text. A variable P will be defined which includes any consecutive values to be the neural network inputs, and the target for these neural networks will be the encrypted text values. The parameters will be set and neural network trained to set weights for the desired values. Third, the resulting weights need to be set and sent to the receiver. At the receiver side, the same neural network will be built, the positions of the encrypted text inside the image will be extracted, using the same encryption algorithm as on the sender side, the original text message will be retrieved. The components for the steganography process at both sides of sender and receiver will be discussed in detail.

The main steps done on both sender and receiver side will be discussed separately and then as a whole process.

3.1.1 Sender Main Steps

- **Original Image:** This stage represents the process of reading the image that would be used. In stage the image representation will be converted into grayscale, the image size used in this project is 300*300 (it can be any size). Here an image is assumed to be a two dimensional matrix and the image is made up of rows and columns of such points known as pixels. At least three different cover images will be used.

Figure 2. Cover image (Lena) Courtesy of [14]

- **Convert the image from two-dimensions to one dimensional array.**

The image will be converted from two dimensions to a one dimensional array (vectorized), which makes searching through it faster. If no changes are made, the process will be more complex.

- **Secret message encryption**

At the same time as the cover image is converted, the secret message will be encrypted by an encryption algorithm. A new algorithm will be utilized, in which each text digit will be mapped to a number using two number ranges, odd numbers (1-81) and even numbers (2-70), for example (A→1 , B→3 , and so on). At this stage, not only English characters will be mapped but also numbers, symbols, and the newest development at this stage is mapping the Arabic characters.

- **Combine each number with its first equivalent position in the image and take the positions**

After the number for each text is obtained it is combined with the first equivalent match. In the cover image, then the position of the first equivalent match will be taken to produce a new array of the equivalent positions.

- **Create The Neural Network**

In this step a neural network of the Radial Basis Function (RBF) was created and applied to the matrix of positions. A variable P was defined which will be any N consecutive values to be the neural network inputs, where N is the length of the text to be encrypted. The target for this neural network will be the matrix of positions.

- **Set The Parameters And Train The NN**

RBF Neural Network used needs some parameters to bet specified, like the spread ∂. Alternatively, it is not changed and remains at its default value (one), and after this the NN will be trained to set the weights to the desired values.

- **Save The Resulted Weights**

Save the resulted weights matrix which will considered as the Cipher text, and sent to the receiver.

3.1.2 Receiver Main Steps

The decryption process done by the receiver is an inverse of what happens at the sender side. Only the person who knows decryption key will be able to see the original text, anyone else will not see anything different in the received image.

- **Load The Weights**

In this step the weights received are loaded which is in fact the cipher text sent by the sender.

- Create the same Neural Network as sender side

At the receiver side the same RBF neural network is created as at the sender side.

- **Set the NN Weights Using The Loaded Matrix**

Then, the created NN using the weights are set which were provided from the received cipher text.

- **Use The Same Input Sequence And Parameters Used By The Sender**

At the receiver side the same input sequence as used by the sender must be loaded, and the parameters must also be the same. If the input sequence or the parameter (∂) values are changed, the the original text cannot be retrieved.

- **Get the positions from weights**

From the weights entered to the neural network the positions can be obtained which are then converted to numbers.

- **Combine each position with its first equivalent pixel value in the image and take the values**

After the position matrix is obtained it is combined with the first equivalent match on the cover image, then the pixel values of the first equivalent match will be taken to produce anew array of the equivalent values.

- **Secret message decryption**

The secret message will be decrypted by the same encryption algorithm used at the sender side. The same algorithm will be used in reverse mode, each number will be mapped to a text digit using two number ranges, odd numbers (1-81)and even numbers (2-70), for example (1→A , 3→B , and so on).

3.3 Security Characteristics

In this section, this project is addressed from the point of security. As previously determined, both sender and receiver must agree on some common values before starting the Encryption/Decryption process. These parameters are:

- **Text length:** Both sender and receiver must know the text length to generate the NN input values, and to know how to break the original\encrypted text.
- **Input sequence:** To generate the same position values, the same inputs values must be used by the sender to generate the weights values.
- **NN characteristics:** Identical NN must be used on both sides, activation function, number of the hidden layers, number of neurons on each layer, other parameters like (spread, center ….) all of them must be the same to guarantee positive results. Also, the previous three parameters, considered to be a private key, must be known by both sender and receiver before starting the Encryption/Decryption process.

4. Experimental Results and Discussions

The proposed method, which is a combination between steganography and cryptography (Encryption), is a process showing how to encode messages in such a way that hackers cannot know of their existence, but only an authorized person can. In the encryption technique, the text is encrypted by the encryption algorithm, to turn it into an unreadable cipher-text. This is done with an encryption key, which specifies how the text is to be encoded.

An encryption algorithm is used to generate new values for the original text, resulting cipher text should be different from the original one so that no person other than the authorized one can see it. The difference between the two texts (original and cipher) should be as large as possible. Also, hidden features of the encrypted text should be more than what the original text has but this is not sufficient, so the encryption algorithm should be evaluated by other metrics.

The metrics that will be used to evaluate the proposed steganography method are: the mean square error, the histogram of text, and the number of pixels change rate (NPCR).

- In this part the experimental results obtained by the steganography algorithm will be presented.
- There are four phases :
  a- Creation of the neural network using RBF function
  b- Using the generated weights to make private keys and the encrypted text
  c- At the receiver side the encrypted text and private key will be compiled to regenerate weights.
  d- These weights are used to decrypt the original text.
- Encryption and Decryption algorithm will be implemented using MATLAB software.
- Three different images were used to test measurement metrics.

4.1 Proposed Method Measurement Metrics

4.1.1 NPCR Results

NPCR will be used to measure the difference between pixels values in both encrypted and original texts O (i,j) and E(i,j) denote the values of pixels in the original and encrypted texts respectively, so NPCR can be defined as following (Khare et al., 2010):

$$NPCR = \frac{\sum_{i=1}^{l} W(i)}{l} \times 100\% \quad (5)$$

Where:

[l]: text length

$$W(i) = \begin{cases} 0, if\ \ E1(i) = E2(i) \\ 1, if\ \ E1(i) \neq E2(i) \end{cases} \quad (6)$$

Where:

[E1]: original text

[E2]: encrypted text

If NPCR equals 100% indicates that both text the original and encrypted are totally different in amplitude.

The NPCR value from matlab code for Lena image is **98.28%**

4.1.2 Mean Square Error

In this project NEWRBE algorithm is used, which generates weights with zero error in the weights values, so that in the decryption side the result text should be 100% compatible with the original one, and the MSE value should be zero. In our test code the results for MSE using three different images are approximately zero (Geetha & Prasad, 2014).

The MSE value from Matlab code for Lena image is **0.1073**

4.1.3 Histograms

The histogram is a frequency analysis which shows how many times a letter appears in the text. The histogram of original text differs from the histogram of the encrypted text and the Figures 3, 4 below show the histograms for secret message using Lena test images and its encrypted text.

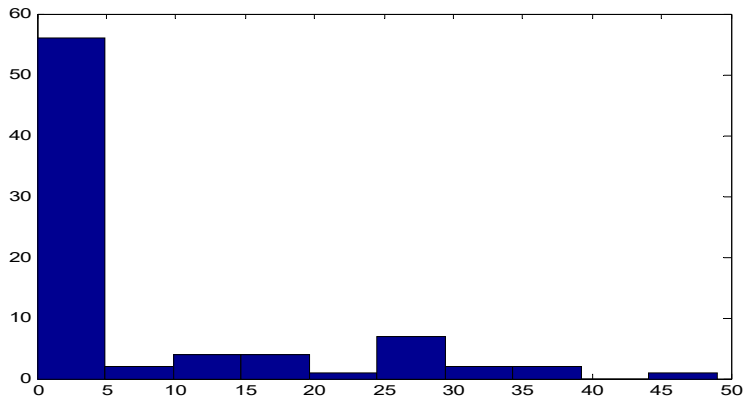

Figure 3. Histogram of original text using Lena image

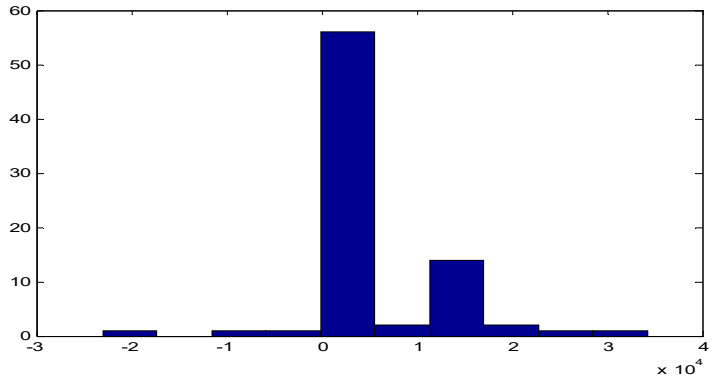

Figure 4. Histogram of encrypted text using Lena image

5. Conclusion

Both steganography and cryptography are excellent methods through which to accomplish secure transmission, but neither method on its own is perfect and both can be broken. Steganographic research is primarily driven by the weakness in the cryptographic systems alone and the desire to have acomplete secret solution in an open-system.

Steganography is used in transmitting texts in a safe way Therefore, the algorithm proposed can send text with high performance, high security, and it can be implemented in real-time applications. The process occurs by applying RBF Neural Networks algorithms. Similarly, the same encoding method is used inversely, for decoding on the transmitting side. RBF has short processing time and a lower error rate. Similarly to the PSNR, MSE values vary based on the amount of data in asecret message.

The main difference of this study to other related ones is not to make any changes or have bit losses in the cover image because the method depends on the similarity between the mapped text number and the image pixel, also when using RBF for encoding and decoding. The procedure of this study consists of decoding the encrypted text by RBF to obtain the original text. As a result, it succeded to obtain almost the same text of the original text easily by RBF (MSE=0).

Finally the histogram can be used to measure the power of the encryption algorithm; the histogram of encrypted text should differ from the histogram of the original text.

Acknowledgements

Thanks to my family, for their love, support and patience during all the preparation of this paper, and all friends who have contributed with helpful and comments during the evolution of this paper.

References

Geetha, B., & Prasad, E. V. (2014). High Secure Im-age Steganography Based On Hopfield Chaotic Neu-ral Network and Wavelet Transforms. *International Journal of Computer Science and Network Security, 14*(3), 93-98.

Gurney, K. (1999). *An Introduction to Neural Networks*. Routledge.

Halici, U. (2001). *Artificial Neural Networks "in-formation institute, middle east technical university*.

Kaur, L., & Geetanjali, B. (2014). Improved Protection in Image Steganography using Neural Network and Discrete Cosine Transform. *International Journal of Application or Innovation in Engineering & Management, 3*(11).

Khare, A., Meenu, K., & Pallavi, K. (2010). Efficient Algorithm for Digital Image Steganography. *Journal of Information, Knowledge and Re-search in Computer Science and Applications, 1*(1), 1-5.

Narayana, S., & Prasad, G. (2010). Two New Approaches for Secured Image Steganography Using Cryptographic Techniques and Type Conversions. *An International Journal (SIPIJ), 1*(2). https://doi.org/10.5121/sipij.2010.1206

Siddharth, S., & Siddiqui, T. J. (2012). A security enhanced robust steganography algorithm for data hiding. *International Journal of Computer Science, 9*(3), 131-139.

Usha, B. A., Srinath, N. K., & Cauvery, N. K. (2013). Data embedding technique in image ste-ganography using neural network. *International journal of advanced research in computer and com-munication engineering, 2*(5), 2319-5940.

Van Tilborg, H. (1999). F*undamentals of Cryptology*. Eindhoven University of Technology, the Netherlands, Kluwer Academic Publishers, Boston/Dordrecht/London.