Image Steganography Based on Discrete Wavelet Transform and Enhancing Resilient Backpropogation Neural Network

إخفاء صورة أعتماداً على التحويل المويجي المتقطع والشبكة العصبية المرنة المحسنة ذات الأنتشار الخلفي

By

Ahmed Shihab Ahmed AL-Naima

Supervisor

Prof. Dr. Reyadh Shaker Naoum

Co-Supervisor

Dr. Sadeq AlHamouz

A Thesis Submitted in Partial Fulfillment of the Requirements of the Master Degree in Computer Science

Department of Computer Science

Faculty of Information Technology

Middle East University

Amman- Jordan

(March-2015)

Authorization Statement

I, Ahmed Shihab Ahmed AL- Naima, authorize Middle East University to supply hard and electronic copies of my thesis to libraries, establishments, bodies and institutions concerned with research and scientific studies upon request, according to the university regulations.

Name: Ahmed Shihab Ahmed AL- Naima

Date: 30/3/2015

Signature: Ahmed shihab

إقرار تفويض

أنا احمد شهاب احمد النعيمي أفوض جامعة الشرق الاوسط بتزويد نسخ من رسالتي للمكتبات أو المؤسسات أو الهيئات المعنية بالأبحاث والدراسات العلمية أو الأفراد عند طلبها.

الاسم: احمد شهاب احمد النعيمي

التأريخ: ٢٠١٥/٣/٣٠

التوقيع: 

Examination Committee Decision

This is to certifying that the thesis entitled "Image Steganography Based on Discrete Wavelet Transform and Enhancing Resilient Backpropogation Neural Network" Was successfully defended and approved on March 30, 2015.

Examination Committee Members **Signature**

(Head of the Committee)

Dr . Maamoun Ahmed
Assistant Professor
Head of Department of Computer Science
Faculty of Information Technology
(Middle East University)

(Member & Supervisor)

Dr . Sadeq AlHamouz
Associate Professor
Chairman of the Computer Information Systems
Faculty of Information Technology
(Middle East University)

(External Committee Member)

Dr . Ezz Hattab
Associate Professor
Department of Computer Science
(Princess Sumaya University for Technology)

DEDICATION

This thesis is dedicated to all the people who never stop believing in me:

My Father

My Mother

My Brother & Sisters

My Wife, who taught me to get up after a fall and start again.

My Children: Ali & Tabarak

ACKNOWLEDGEMENTS

First and foremost I would like to express my heartfelt gratitude and appreciation to my supervisor **Prof. Dr. Reyadh Naoum** who without his meticulous supervision, faithful guidance and continuous support, this work could never be accomplished.

Also, I wish to extend my profound gratitude and thanks to my Co-Supervisor **Dr. Sadeq AlHamouz** and the **Dean of the faculty of information technology in Middle East University** and all the teaching staffs for every bit they have done to me.

Genuine thanks all my friends for their help and support.

Finally, I would like to express my love and gratitude to my family for their support and patience over this adventure. I don’t think I would’ve been able to do this without them.

Table of Contents

Image Steganography Based on Discrete Wavelet Transform and Enhancing Backpropogation Neural Network I
Authorization Statement II
إقرار تفويض III
Examination Committee Decision IV
DEDICATION V
ACKNOWLEDGMENT VI
Table of Contents VII
List of Tables X
List of Figures XI
List of Abbreviations XIII
Abstract XIV
الملخص XVI

Chapter (1) : Introduction 1
1.1 Introduction 1
1.2 Problem Statement 3
1.3 Objectives of the Thesis 4
1.4 Motivation 5
1.5 Methodology 5
1.6 Published Work 6
1.7 Thesis Outline 6

Chapter (2) : Literature Review 7
2.1 Literature Review 7
2.1.1 Image Steganography based on Spatial Domain: Image in Image 7
2.1.2 Image Steganography based on Transform Domain: Data in Image 8
2.1.3 Image Steganography based on Transform Domain: Image in Image 10

Chapter (3) : Principles and Fidelity Criteria of Steganography 15
3.1 Introduction 15
3.2 Characterization of Steganography Systems 16
3.2.1 Invisibility (Perceptual Transparency) 16
3.2.2 Security 16
3.2.3 Undetectability 16
3.2.4 Robustness 17
3.2.5 Capacity 17
3.3 General Steganography Framework 17
3.4 Steganography Methods 18
3.4.1 Steganography in Text 18

3.4.2 Steganography in Image 19
3.4.3 Steganography in Video 19
3.4.4 Steganography in Audio 20
3.5 Steganographic Techniques 20
3.5.1 Substitution Systems 20
A. Least Significant Bit (LSB) Substitution 21
B. Pseudorandom Permutation 22
3.5.2 Transform Domain Techniques 22
A. Wavelet Transform 23
3.6 Discrete Wavelet Transform 24
3.6.1 Haar- Discrete Wavelet Transform 24
3.6.2 Two-Dimensional Haar-Discrete Wavelet Transform 25
3.7 Steganography Attackers 27
3.7.1 Passive Attacker 27
3.7.2 Active Attacker 28
3.7.3 Malicious Attacker 28
3.8 Countermeasures Against Attacks 28
3.9 Fidelity Criteria 29
3.9.1 Peak Signal -to- Noise Ratio (PSNR) 29
3.9.2 Mean Square Error (MSE) 30

Chapter (4) : Artificial Neural Networks 31
4.1 Introduction 31
4.2 The Neural Network Mathematical Model 32
4.3 Architecture of Neural Network 33
4.4 The Learning Process 35
4.4.1 Supervised Learning 35
4.4.2 Unsupervised Learning or Self- organization 36
4.5 Architectures for Training of an Artificial Neural Network 37
4.5.1 Back-propagation Neural Network 38
4.5.2 Adaptive Back-propagation Learning : The (RPROP) Algorithm 39

Chapter (5): Proposed Artificial Neural Network- Steganography System ..... Implementation 42
5.1 Introduction 42
5.2 Pre-Embedding stages 43
5.2.1 Secret Image Selection and Processing Stage..... 43
A. Secret Image (RED, Green, Blue) Splitting 44
B. Discrete Wavelet Decomposition of Secret Image 44
C. Conversion of Secret Image Sub bands to Bit Streams..... 45
D. Key Generation and bit Streams Encryption 46
5.2.2 Best Cover Image Selection and Processing Stage 47
A. Enhanced Resilient Back-propagation Algorithm Training 49

B. Best Learning Parameter (ξ) to Enhance (RPROP)..... 50
C. Best Cover Image (RED, Green, Blue) Splitting 51
D. Discrete Wavelet Decomposition of best Cover Image..... 51
5.2.3 Best Embedding Threshold Selection Stage 53
5.3 Embedding Phase 55
5.4 Embedding Process 61
5.5 Extraction Phase 64
5.6 Extraction Process 65

Chapter (6): Experimental Results, Conclusion and Future Work 69
6.1 Implementation 69
6.2 Experimental Results of the Proposed System 69
6.2.1 Experimental Results of the Embedding Phase 71
6.2.2 Experimental Result of the Extraction Phase 79
6.3 Processing Time Comparison between Original Proposed Embedding Model and Modified Proposed Embedding Model 86
6.4 Comparing our proposed algorithm with other algorithms 87
6.5 Conclusions 88
6.6 Future Work 89
References 90

List of Tables

Table (5.1): (ERPROP) neural networks parameters ..……..... 50
Table (6.1): The PSNR and MSE values of case study (1) 71
Table (6.2): The PSNR and MSE values of case study (2) .....…….. 73
Table (6.3): The PSNR and MSE values of case study (3) 74
Table (6.4): The PSNR and MSE values of case study (4) 76
Table (6.5): The PSNR and MSE values of case study (5) 77
Table (6.6): Image size (in bytes) comparison between cover and stego images 79
Table (6.7): The PSNR and MSE values for secret image extraction case study (1) 80
Table (6.8): The PSNR and MSE values for secret image extraction case study (2) 81
Table (6.9): The PSNR and MSE values for secret image extraction case study (3) 82
Table (6.10): The PSNR and MSE values for secret image extraction case study (4) ... 83
Table (6.11): The PSNR and MSE values for secret image extraction case study (5) ... 85
Table (6.12): PSNR and Processing time of proposed embedding models (original and modified)..... 86
Table (6.13): PSNR of our proposed method and DWT method 87
Table (6.14): PSNR of our proposed method and DWT method 87
Table (6.15): PSNR of our proposed method and DCT method 87

List of Figures

Figure ..... Page
Figure 1.1: Information-hiding system features 2
Figure 3.1: General steganography framework 18
Figure 3.2:LSB in 8 bits per sample signal is overwritten by one bit of the hidden data 20
Figure 3.3: 1-Level 2D –DWT (a) Decomposition, (b) Reconstruction 26
Figure 3.4: 1-Level 2D- DWT decomposition of an image 27
Figure 4.1: The neural network mathematical model 32
Figure 4.2: Architecture of neural network 34
Figure 4.3: Sigmoid transfer function 35
Figure 4.4: Overview of supervised learning 35
Figure 4.5: The unsupervised learning process 36
Figure 5.1: Secret image selection and processing stage block diagram 43
Figure 5.2: RGB layers separation of secret image 44
Figure 5.3: 1-level decomposition of the full color (RGB) secret image 44
Figure 5.4: 1-level decomposition of Red, Green and Blue layers of secret image 45
Figure 5.5: Bit stream conversion of secret image coefficient of red layer 46
Figure 5.6: A 8-bit modified fibonacci linear feedback shift register 46
Figure 5.7: Best cover image selection and processing using hybrid of (SOM) and (ERPROP) 48
Figure 5.8: RGB layers separation of best cover image 51
Figure 5.9: 4- Level Haar- DWT decomposition of the full color (RGB) cover 52
Figure 5.10: 4-Level decomposition of Red, Green and Blue layers of cover image ... 52
Figure 5.11: Best embedding threshold selection stage block diagram 53
Figure 5.12: Enhanced resilient back propagation training process to select best embedding threshold block diagram 54
Figure 5.13: Approximation sub band of secret image embedding in the approximation sub band of cover image 55
Figure 5.14 (a): Example depicting the embedding operations of CAr of secret image in CAr of cover image 57
Figure 5.14 (b): Example depicting the operations steps of LSB substitution 57
Figure 5.15: Original proposed embedding model 58
Figure 5.16: The modified proposed embedding model 60
Figure 5.17: Embedding algorithm flowchart of embedding red color layer of secret image in the corresponding red color layer of cover image 63
Figure 5.18: Proposed extraction model 64
Figure 5.19: Depicting the extraction algorithm flow chart of extracting red color layer of secret image from the corresponding red color layer of stego image 67
Figure 5.20: Proposed model using statistical equation (5.1) 68

Figure 6.1: Shows the selected secret images and the corresponding best covers images chosen by (ERPROP) 70
Figure 6.2 (a) : (64x64) Bird 2 secret image 71
Figure 6.2 (b) : Flower cover image with its histogram 72
Figure 6.2 (c) : Flower stego image with its histogram 72
Figure 6.3 (a) : (100x100) Chinook secret image 73
Figure 6.3 (b): Sydney City cover image with its histogram 73
Figure 6.3 (c): Sydney City stego image with its histogram 74
Figure 6.4 (a): (128x128) Tomahawk secret image 75
Figure 6.4 (b): Baboon cover image with its histogram 75
Figure 6.4 (c): Baboon stego image with its histogram 75
Figure 6.5 (a): (128x128) Bird 1 secret image 76
Figure 6.5 (b): Garden cover image with its histogram 77
Figure 6.5 (c): Garden stego image with its histogram 77
Figure 6.6 (a): (150x150) F-16 secret image 78
Figure 6.6 (b): Baboon cover image with its histogram 78
Figure 6.6 (c): Baboon stego image with its histogram 78
Figure 6.7 (a): Original secret image with its histogram 80
Figure 6.7 (b): Extracted secret image with its histogram 80
Figure 6.8 (a): Original secret image with its histogram 81
Figure 6.8 (b): Extracted secret image with its histogram 82
Figure 6.9 (a) Original secret image with its histogram 82
Figure 6.9 (b) Extracted secret image with its histogram 83
Figure 6.10 (a) Original secret image with its histogram 84
Figure 6.10 (b) Extracted secret image with its histogram 84
Figure 6.11 (a): Original secret image with its histogram 85
Figure 6.11 (b): Extracted secret image with its histogram 85

List of Abbreviations

ERPROP	Enhanced Resilient Backpropagation Neural Network
DWT	Discrete Wavelet Transform
JPEG	Joint Photographic Expert Group
RGB	Red, Green and Blue
BMP	Microsoft Windows Bitmap
DCT	Discrete Cosine Transform
SSB-4	System of Steganography using Bit 4
IWT	Integer Wavelet Transform
HSV	Hue-Saturation-Value
NSCT	Non sampled Counterlet Transform
ASCII	American Standard Code for Information Interchange
PCRSMT	Performance Comparison of Robust Steganography Based on Multiple Transformation Techniques
MSB	Most Significant Bit
LSB	Least Significant Bits
CTT	Contourlet Transform
FDSZT	A Frequency Domain Steganography using Z Transform
HDLS	Hybrid Domain in LSB Steganography
FFT	Fourier Transform
PSNR	Peak Signal -to- Noise Ratio
HH	High- High
HL	High- Low
LH	Low -High
LL	Low-Low
RDWT	Redundant Discrete Wavelet Transforms
PSNR	Peak Signal -to- Noise Ratio
MSE	Mean Square Error
ANN	Artificial Neural Network
SOM	Self-Organizing Map
RPROP	Resilient Backpropagation Neural Network
rT	Red Threshold
gT	Green Threshold
bT	Blue Threshold
FLFSR	Fibonacci Linear Feedback Shift Register
XOR	Exclusive OR Gate
OR	OR Gate
CAr	Coefficient Approximation Red
CHr	Coefficient Horizontal Red
CDr	Coefficient Diagonal Red
CVr	Coefficient Vertical Red
T	Threshold
IDWT	Inverse Discrete Wavelet Transform
MATLAB	Matrix Laboratory

Image Steganography Based on Discrete Wavelet Transform and Enhancing Resilient Backpropogation Neural Network

By

Ahmed Shihab Ahmed AL-Naima

Supervisor

Prof. Dr. Reyadh Shaker Naoum

Co-Supervisor

Dr. Sadeq AlHamouz

Abstract

Steganography is the science and art of covert communication. it allows the undercover information transmission and conceal the existence of message itself in content such as video, audio, or image to protect the transmitted information from intruders and unwanted recipients. in past decade, a variety of researches have been conducted on various steganographic schemes in both spatial and transform domain.

In this research, a novel image steganography system that hides both encrypted color image and secret key inside another color cover image was proposed using a combination of Discrete Wavelet Transform Technique (DWT) and Enhanced Resilient Backpropagation Neural Network (ERPROP).

In this research, we apply the DWT for all color layers (Red, Green, and Blue) separately for both cover and secret image with different levels; 1-level for the secret image and 4-level for cover image where encrypted coefficients of sub bands of the secret image are embedded in the corresponding sub bands of the cover image.

The enhanced resilient backpropagation neural network is applied in two stages. The first stage is to choose the best cover image that will be used to conceal the secret image, while the second stage is to choose the best embedding threshold that will be used to determine the embedding locations in both embedding and extraction phases.

This research, takes advantage of combination between cryptography and steganography to enhance the security and robustness of the system where the sub bands of the secret image converted to bit streams and then encrypted before the embedding phase.

This system was implemented using MATLAB 7.14, R2012a, and it is proven to be pre-eminence, in comparison with other existing steganographic systems in terms of Peak Signal-to- Noise Ratio (PSNR) and capacity. The PSNR of embedding phase reaching up to (112.4780) dB and the secret image is recovered with PSNR reaching up to (93.1047) dB. These satisfactory results are fulfilled in combination with multilayer security. Therefore, our proposed system achieved the steganographic goals that was built for.

إخفاء صورة أعتماداً على التحويل المويجي المتقطع والشبكة العصبية المرنة المحسنة ذات الأنتشار الخلفي

إعداد

أحمد شهاب أحمد النعيمي

إشراف

أ.د. رياض شاكر نعوم

المشرف المشارك

د. صادق الحموز

الملخص

إخفاء المعلومات علم وفن في الاتصالات السرية. يسمح هذا العلم بنقل المعلومات السرية وإخفاء وجود الرسالة نفسها في محتوى مثل فيديو أو صوت أو صورة ليحمي المعلومات المنقولة من المتطفلين او المستقبلين غير المرغوب بهم. في العقد الماضي، العديد من الأبحاث أجريت على طرق مختلفة في علم اخفاء المعلومات في كلا المجالين: المجال المكاني والمجال التحويلي.

في هذا البحث، تم اقتراح نظام جديد لإخفاء الصوُر، ويخفيِ كلاً من الصورة الملونة المشفرة والمفتاح السري داخل صورة غطاء ملونة أخرى باستعمال تقنية التحويل المويجي المتقطع والشبكة العصبية المرنة المحسنة ذات الانتشار الخلفي.

في هذا البحث، نستعمل تقنية التحويل المويجي المتقطع لجميع طبقات الألوان (أحمر، أخضر، وأزرق) بشكل منفصل لكلٍ من صورة الغطاء والصورة السرية بمستويات مختلفة؛ مستوى واحد للصورة السرية وأربع مستويات لصورة الغطاء حيث يتم تضمين المعاملات المشفرة في المقاطع الفرعية للصورة السرية داخل المقاطع الفرعية المقابلة لها في صورة الغطاء.

تطبق الشبكة العصبية المرنة المحسنة ذات الانتشار الخلفي على مرحلتين: الأولى هي اختيار أفضل صورة غطاء سيتم استعمالها لإخفاء الصورة السرية، أما الثانية فهي اختيار أفضل عتبة تضمين سوف يتم استعمالها لتحديد مواقع التضمين في مرحلتي التضمين والاستخراج.

هذا البحث يأخذ بمزايا الجمع بين استعمال التشفير وإخفاء المعلومات لتعزيز أمن ومتانة النظام حيث تتحول المقاطع الفرعية للصورة السرية إلى سلسلة ارقام ثنائية (bit streams) ومن ثمَّ تشفيرها قبل مرحلة التضمين.

يتم تنفيذ هذا النظام باستعمال MATLAB 7.14, R2012a وقد أثبت تفوقه العالي مقارنةً بأنظمة الإخفاء الحالية الأخرى من حيث نسبة قمة الإشارة إلى الضوضاء (PSNR) والسعة. وتصل نسبة قمة الإشارة إلى الضوضاء لمرحلة التضمين (112.4780) ديسيبل وتسترد الصورة السرية بنسبة قمة إشارة إلى ضوضاء تصل (93.1047) ديسيبل. تتحقق هذه النتائج المرضية مع تحقيق أمن متعدد الطبقات. وعليه، يحقق نظامنا المقترح أهداف إخفاء الصورة التي صمم النظام من أجل تحقيقها.

Chapter One

Introduction

1.1 Introduction

The concept of information hiding is ancient technique which is traced back to a thousand years ago. It is merely based on dimming messages content by a process called encryption, which is sometimes not practically effective. In many competitive cases, it is highly demanded to suppress the initial existence of a communication in order to avoid suspicion from adversaries (Wu & Liu, 2003).

Recently, the information hiding techniques have become an important practice in a wide areas and applications, including digital audio, video, and pictures which are equipped dramatically with imperceptible marks that possibly contain a hidden copyright notice, and a serial number, or even have the ability to directly assist in preventing an unauthorized copying process. The military communications systems employ a high level of traffic security techniques which instead of just hiding the message content by the encryption process; it seeks to conceal the message sender, and its receiver, or even its very existence (Petitcolas, et al., 1999).

The "information hiding" term is linked to both steganography and digital watermarking. The *steganography* is defined as the attempt to hide the fact that information is being transmitted at the first place, while *watermarking* is usually referred to the involved methods by which an identified information is being hidden in a data object, and accordingly, the information will be kept robust against modification (Stamp, 2006).

Steganography is considered as a kind of a hidden communication which means literally the “covered writing”. Originally, it has been derived from the two Greek words stegano which refers to “covered” and graphos which refers to “to write”. The goal of applying the steganography is to hide the information message inside a harmless cover medium by a certain way that makes it impossible to detect the secret information and even its existence in the cover medium (Parah et al., 2012).

Data hiding is a form of steganography that works by embedding data into digital media for the purpose of identification, annotation, and copyrighting. In fact, several manacles affect such a process and this is including, the quantity of the proposed data to be hidden, the need for invariance of these data under conditions where a “host” signal is subjected to distortions, e.g., lossy compression, in addition to the degree to which the data must be immune to an interception, a modification, or a removal by a third party (Bender et al., 1996).

The three main aspects regarding information hiding systems are *capacity*, *security* and *robustness* as represented in figure 1.1. Generally, steganography requires a high security and capacity levels, i.e. hidden information are usually a fragile ones which can be destroyed by even slight modifications. On the other hand, watermarking mostly relies on achieving a robustness status where it is impossible to remove the watermark without causing severe cover content quality degradation (Sridev et al., 2011).

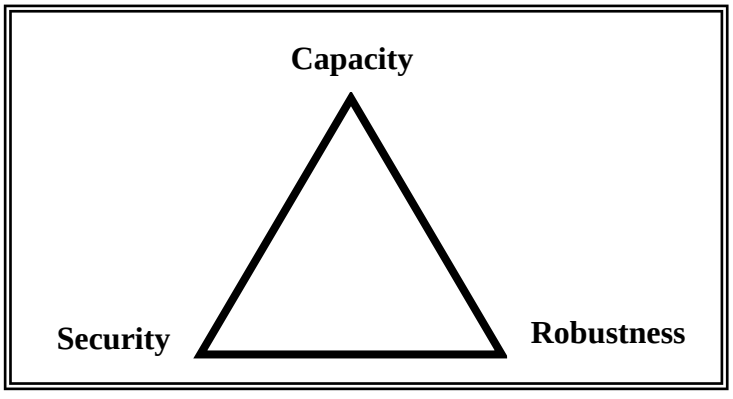


Figure 1.1: Information-hiding system features.

Steganography is recognized as a close technology to cryptography. Both have been employed to add elements of secrecy within a communication process. Cryptographic techniques work by scramble a message, so in case it is intercepted, it will not be understood. Practically, this process is known as an encryption where an encrypted message is usually referred to as a cipher text. On the other hand, the core work of a steganography is through gilding a message in an aim to hide its original existence and turn it into an invisible one, thus entirely hiding the fact that a message has being sent. Interestingly, the ciphertext message is may be able to draw a suspicion, while an invisible message is incapable of doing that (Johnson et al., 2001).

1.2 Problem Statement

The threat of an individual or a group misusing the normal ways of communications to steal information sent between people, or jamming on specific information communicated between nations is more likely now than ever. Due to the nature of these types of communications, the need for a means to hide information and keep it secure is essential. In fact, the available traditional information hiding techniques cannot meet the challenges of either large-scale or highly sophisticated network attacks (Narasimmalou & Joseph, 2012).

Therefore, in response to the above problem, our research proposes a novel modern steganography system that combine discrete wavelet transform, cryptography and enhanced resilient back propagation neural networks for transmitting pictures with a multilayer of security and robustness that give the system the capability to deal with large-scale complicated network intrusions and meets the need for a further better perceptual transparency against attackers during information transmission.

The main questions in this research are identified as follows:

- How to train the Enhanced Resilient Back-Propagation (ERPROP) for selecting the best cover image to embed the secret image?
- How to train the (ERPROP) to select the best embedding threshold for each color layer?
- Is it possible to embed a secret image inside cover image, using (DWT) and (ERPROP), without significant change in the quality of the stego image?
- Is it possible to reduce image distortion and improving undetectability by using Discrete Wavelet Transform (DWT)?
- Is it possible to increase the robustness and security by using the (DWT) and (ERPROP)?

1.3 Objectives of the Thesis

The following is the main objectives of this thesis:

1. To hide full color secret image inside full color cover image without affecting them perceptually in a way that can avoid drawing any suspicion to the stego-image.
2. To Propose a new robust and secure steganography system, based on a combination of two powerful algorithms (DWT and ERPROP).
3. To show that encrypting the secret image and hiding it inside the stego image add another layer of protection, robustness and enhance the proposed system performance.
4. To show that the proposed system is able of extracting embedded image efficiently.
5. To show that the new steganographic technique incorporates a high level of robustness and grants a high level of secrecy to resist the extraction process which is done by attacker.
6. To maintain the stego image high perceptual transparency as well as the high data hiding capacity of the cover image.

1.4 Motivation

Recently, the internet usage is growing rapidly over high bandwidth and millions of low cost computer hardware's, such an explosive growth in data communication is associated with a major problem of ensuring secure transmission and preventing unauthorized accesses. Particularly, in the military applications and communications between members of companies and organizations, hiding secret information has an essential importance in securing one to one communication. Therefore, great efforts have been made to establish ways of transmitting data with a high level of security.

Steganography is one of powerful ways that has been used for information security; it allows both hiding the content of the information that to be transferred as well as hiding its existence from the eyes of intruders which is exactly what people nowadays are seeking through their internet communication.

1.5 Methodology

In this research work, we are considering the following aspects:

1. Reviewing the literatures about the back propagation artificial neural networks and steganography at different domains, and particularly focusing on the discrete wavelet transform domain which is selected as the core of our proposed steganography system.
2. Developing and implementing a steganography system which is based on the combination of (DWT) and (ERPROP).
3. Experimenting the system operation using many different RGB images (JPEG) as both cover images and secret images with different sizes.
4. Visual and statistical evaluation of the system performance.
5. Discussing the results, give conclusions and suggest recommendations for future work.

1.6 Published Work

- Naoum, R., Shihab, A., AlHamouz, S. (2015). Enhanced Image Steganography System based on Discrete Wavelet Transformation and Resilient Back-Propagation. *International Journal of Computer Science and Network Security, 15* (1).
- Naoum, R., Shihab, A., AlHamouz, S. (2015). A novel Image Steganography System based on Hybrid Artificial Neural Network with Haar-Discrete Wavelet Transform. *Indian Journal of Applied Research, 5* (3).
- Naoum, R., Shaker, M., Mudhafar.J., & Shihab. A. (2014). Discrete Wavelet Transform for Image-to-Image Steganography. *European Journal of Scientific Research, 117* (1).
- Naoum, R., AlHamouz, S., Shihab, A. & Shaker, M. (2014). Image Steganography using Three Layers DCT and Artificial Neural Network. *European Journal of Scientific Research, 121* (3).
- Naoum, R., Viktorov, O., Shihab, A., & Shaker, M. (2013). Image-to-mage Steganography Based on Discrete Cosine Transform. *European Journal of Scientific Research, 106* (4).

1.7 Thesis Outline

The rest of our thesis is organized as follows: Chapter two presents the various algorithms proposed on the field of steganography. Then, the principles and fidelity criteria of steganography, including the concepts of characterization of steganography system, general steganography frameworks are presented in Chapter three. Chapter four presents different neural network topologies and architectures with special emphasis on back propagation and resilient adaptive back propagation neural networks. The fifth chapter presented our proposed system implementation in terms of its building modules. Finally, Chapter six discuses our proposed system experimental results and summarizes the deduced points, recommendations, and future works.

Chapter Two

Literature Review

2.1 Literature Review

During the last decade, the digital Image steganography method is believed to be one of the image processing domains which are dramatically on demand, and that is due to the digital multimedia technologies momentary development. In this section, we are elucidating an overview of what have been used in previous research works in an aim to abstract the prime employed techniques and methodologies, while the following literature surveys are driven from a recognized published works that aims to describe previous research and development deeds that has been accomplished on digital steganography method.

2.1.1 Image Steganography based on Spatial Domain: Image in Image

Al-Jbara et al. (2012) proposed a new approach of image steganography that hides a gray scale secret image type into a colored (BMP) image type. Such an approach includes two phases, the first phase is where the secret image is compressed by the application of the artificial neural network technique (Back-propagation algorithm), while the second phase is where the secret image hiding takes place through using the least significant bit substitution method. Upon merging these two techniques, the described approach would achieve an elevated hiding capacity of an image data without any noteworthy defacement on the cover image, which reached up to 88.8906% of the cover image size.

2.1.2 Image Steganography based on Transform Domain: Data in Image

Kafri & Suleiman (2009) used a combination of transform/frequency domain which involves Discrete Cosine Transform (DCT) and the notion/data of SSB-4 technique of spatial domain steganography. The main idea of this technique is to use significant bit (4^{th} bit) of the DCT coefficients of cover image in order to hide the bits message. This approach modifies the 4^{th} bit of the coefficients while retaining the minimum difference between the original value and the modified one. The obtained experimental result indicate that, the embedding information in the main significant bit of the DCT domain, the hidden message occupied more robust areas, and provide better level of resistance against steganalysis process.

Lesly & Roy (2012) offered a novel data hiding technique in digital images. Briefly, they employed an adaptive hiding capacity function along with a Kohonen neural network algorithm. Practically, this method embed secret data at the integer wavelet coefficients (IWT). As for the Kohonen network, it was trained according to the absolute contrast sensitivity of the pixels reside in the cover image. The wavelet coefficients are classified into largest, smallest and rest class depending on the contrast sensitivity of each pixel. Each of the selected coefficients is able to hide a different number of secret message bits. This adaptive system is a typical solution for lifting the hidden capacity without degradation the visual quality of the final stego image.

Vani & Prasad (2013) designed a high secure steganography algorithm using (DWT) and Hopfield Chaotic Neural network. The proposed system consisted of three stages: first where the text is encrypted by applying a traditional encryption method named the *Caeser* method, second, the cipher text is encrypted for a second time by employing the

chaotic neural network and ,third where the encrypted text is embedded by using DWT inside the high frequency components of a cover image of the gray scale type.

Alsaif & Salih (2013) developed a new data hiding technique for embedding text data in images represented in Hue-Saturation-Value (HSV) color space model and by using a non-sub sampled contourlet transform (NSCT). First, the text data is turned into an ASCII format in order to be represented in a binary form, so later it can be added to contourlet coefficients. A high frequency directional pass band is selected from the contourlet transform for the purpose of data embedding. Moreover, by applying this method, a higher capacity can be achieved depending on the block size and with different threshold values. From the experimental results of applying the proposed idea of the contourlet coefficient, shows that hiding data in the coefficients of the contourlet gives a robust technique for high security.

Nain & Bansal (2014) suggested merging a text into a color cover image and based on employing Block-DCT vector quantization method, where the DCT is applied to transform original image (cover image) blocks from a spatial domain into a frequency domain. At such case, the artificial neural network can assist in finding the pixels in order to merge the data bits without much impacting the original pattern. This technique is based on least significant bits replacement using the DCT coefficient pixel values. Researcher found that employing neural network system in a combination with the DCT method can boost up the image capacity to hide certain kind of messages, and enhanced the quality of stego image.

2.1.3 Image steganography based on Transform Domain: Image in Image

Kumar et al. (2011) implemented a Performance Comparison of Robust Steganography Based on Multiple Transformation Techniques (PCRSMT). Both of the cover image and the payloads were applied along with implementation of DWT and IWT. Most Significant Bits (MSB) of payload IWT coefficients are hidden in the Least Significant Bits (LSB) of IWT coefficients of cover image, to extract the stego-image used for secure data communications. The experiment shows that their algorithm provides higher level security and robustness.

Kumar & Muttoo (2011) developed a steganography technique based on Contourlet Transform (CTT). The proposed technique apply a self-synchronizing variable length code in an aim to encode the secret image. Later, this secret image is embedded in the high frequency sub-bands obtained by applying CTT to the gray-scale of cover-image using rightmost of LSB method and thresholding technique. Final experimental results evidence that the contourlet transform is more applicable for data hiding purposes, as more data can be hidden in the high frequency sub bands without distorting the stego image perceptibility.

Mandal (2011) presented a novel embedding approach termed as, FDSZT based on Z-transformation for gray scale images which were based on the concept of median that has been used to select the coefficient for embedding in Z-Transformed domain. One bit of the secret image was inserted into the cover image byte with a 2×2 mask, and the insertion took place at the rightmost fourth LSB bit of the byte cover image. This technique provided a reasonable integrity for various types of images that adapted Z-transformed steganography, and accordingly such images will gain a preferable visibility and quality.

Kumar, Raja & Pattnaik (2011) suggested a hybrid steganography (HDLS) which is an integration of both transform and spatial domains. each of the cover image and the payload were splitted into two cells. The components of RGB color channels of cover image cell I were first separated and then transformed individually from a spatial domain into a frequency domain applying DCT/DWT/FFT where it embedded in a special approach. on the other hand, the components of cell II of cover image are being retained in spatial domain itself. In the embedding process, four MSB bits of each pixel in the payload cell-1 and cell-2 are hidden in the second and fourth LSB places of cover image cell I and cell II respectively to raise the security of the payload and generate stego image in transform domain.

Singh & Siddiqui (2012) proposed a new robust steganography algorithm based on discrete cosine transform (DCT), Arnold transform and chaotic system. Random sequence is generated by the chaotic system in aim to spread data in the middle frequency band DCT coefficient of the cover image. The security has gone under further enhancement through scrambling the secret image by applying Arnold cat map before embedding process. The experimental results demonstrate that the proposed algorithm reaches to multilayer of invisibility, security and robustness levels against JPEG compression.

Narasimmalou & Joseph (2012) designed a novel steganographic algorithms based on (DWT) and specific embedding equation. Two different approaches were proposed. The first approach applied three level DWT decomposition employing green color layer of the cover image for embedding and then the image was partitioned into swapped 4x4 blocks. The second approach used one level DWT decomposition. The proposed model

attained both high PSNR and perceptuality compared to existing steganographic schemes.

Bhattacharya, Dey & Chaudhuri (2012) developed a Steganographic technique for hiding multiple secret images in a color cover image based on DCT and DWT transforms. The cover image is splitted into four sub-bands by applying DWT. While, the DCT is separately applied in each HH band to get corresponding DCT coefficients. Then, Secret binary images are scattered among the selected DCT coefficients using a pseudo random sequence and a session key. In this approach, embedding is taken place randomly in the frequency domain and as a result it will be difficult to detect the existence of the secret image using classic steganalysis techniques.

Singh & Siddiqui (2012) developed a robust image steganography technique based on redundant discrete wavelet transforms (RDWT). The proposed method consists of two phases: embedding and extraction. In embedding phase, the cover image is partitioned into 8×8 sub blocks and transformed using RDWT. The payload bit was spread using two chaotic sequences, one for ‘0’ and another for ‘1’, each of length equal to the block size of (RDWT). In the extraction phase, the correlation between (RDWT) coefficient and chaotic sequence for '0 ' and '1' bit was found. The combination of chaotic sequence and (RDWT) assisted in achieving enhanced security and robustness against various signal processing attacks along with maintaining high perceptual quality.

Hemalatha et al. (2013) implemented a novel image steganography technique to hide both secret image and key in color cover image using Discrete Wavelet Transform (DWT) and Integer Wavelet Transform (IWT). Actually, the secret image can be regenerated without the need for storing the image itself, instead, an encrypted key is

generated and then the key bits are embedded in least significant bits of (IWT). This technique elevated both the security and the capacity levels.

Naoum, et al. (2013) proposed of hiding color images in another color images. It apply the transform domain techniques within the steganography process in an aim to boost its robustness against any sort of changes and/or treatments done on the cover image. In this research, the DCT employed a smart block matching method between the embedded image and the cover image so as to find the proper locations for hiding the information blocks. However, applying block matching by the DCT method has some concerns and obstacles despite of its great aptitude to retain the embedded information blocks.

Parul, Manju & Rohil (2014) offered a neoteric approach for image steganography applying DWT. The cover image is sectioned into higher and lower frequency sub-bands while secret data was transformed through employing Arnold transformation to boost the security. In the proposed approach, the secret image was partitioned in RGB components and embedded into HL, HH, and LH sub band of RGB respectively. This approach found to be a superior one in terms of PSNR and high embedding capacity.

Naoum, et al. (2014) proposed contemporary study to hide a color image inside another larger color image. The research aimed to use transformation domain methods in order to deal with issues like complexity, security and robustness. The new steganography scheme used a discrete wavelet transform (DWT) method in order to provide better imperceptibility, in harmony with the human visual system, in addition to achieve higher robustness against signal processing attacks. The DWT method was carried out for storing the embedded image in the cover image, whereas an embedding threshold was used to find the hiding locations.

Nitin, et al. (2014) presented a novel image steganography method that was done based on LSB and DCT coefficients that provide randomly scattered bits embedding directly inside the cover image. At first, the Discrete Cosine Transform (DCT) was applied on the cover image and then the secret image was hidden in LSB of the cover image in random locations based on an embedding threshold value. Then, the randomized pixel locations that used to embed secret information were found using DCT coefficients. The whole performance evaluation of the algorithm showed an improvement on both the security and the invisibility of stego image.

Vijay & Vignesh (2014) proposed different work where the (IWT) is carried out on a gray level cover image and in turn the secret image bit stream was embedded into the LSB's of the integer wavelet coefficients of the cover image. The main intent of this work was to concern evolving the embedding capacity and reduce the distortion occurring to the stego image. The (IWT) mapped integers to integers in the area of image steganography which allowed the embedded message to be extracted without loss. The experimental results proved that the evaluation metric such as PSNR is raised in a high manner and the algorithm has both a high capacity and a good invisibility.

Naoum, et al. (2014) developed a novel method to hide a three layer image using Discrete Cosine Transforms (DCT). Resilient Back-Propagation Artificial Neural Network was used as classifier in order to speed up the hiding process via the DCT features. DCT is applied to reduce the redundancy of image information. DCT is also used to embed the secret image, particularly, in the least significant bits of RGB image. Discrete Cosine Transforms works by changing each bit of the secret image in the cover image, only to the extent that is not seen by the eyes of human. The proposed method is implemented and the results show significant improvements.

Chapter Three

Principles and Fidelity Criteria of Steganography

3.1 Introduction

Steganography can be defined as the art of hiding and transmitting data through conspicuous and innocuous carriers in an aim to conceal the existence of the data. Although steganography is considered as an antique craft, yet the proem of computer technology has granted it a second fresh life. In fact, computer-based steganographic techniques have introduced many changes to digital covers to hide outlandish information into the native covers. Such datum could be a communicative one in the form of a text, binary files, or for supplying further information regarding the cover itself (Johnson et al., 2001).

Moreover, the outstanding characters of steganography allowed it to own its place within security fields for the intentions of supplementing the cryptography instead of replacing it. Hiding a message through applying steganography methods can reduce the chance of detecting that message, and in case the message is encrypted, then this would provide an additional layer of security (Taqa et al., 2009). Therefore, some steganographic techniques are combining traditional cryptography along with steganography; where the sender will encrypt the secret message prior to the embedding process. As a matter of fact, such an amalgamation increases the security levels of the total communication process, as it is much complicated for an attacker to reveal the embedded ciphertext in a cover (Katzenbisser & Petitcolas, 2000).

3.2 Characterization of Steganography Systems

Basically, steganographic techniques act by embedding a message inside a cover and several features have to be measured in order to determine the strengths and the weaknesses points of each method, accordingly the relative significance of each feature is depending on its applications.

3.2.1 Invisibility (Perceptual Transparency)

This connotation is based on the attributes of either human visual system or human audio system. The embedded information is imperceptible as long as an average human subject is powerless to differentiate between carriers that do carry hidden information and those that do not. It is essential that the embedding occurs without a significant retraction or a loss in the perceptual quality of the cover (Islam et al., 2010).

3.2.2 Security

It is defined that the embedded algorithm is secure if the embedded information is not subjected to an elimination after being revealed by an attacker, as it depends on the total information about the embedded algorithm as well as the secret key (Vijayakumar, 2011).

3.2.3 Undetectability

The embedded information is considered of an undetectable nature only if the image with the embedded message is harmonious with the source model from which the cover images were initially taken. For instance, if a steganographic method applies the noise component of the digital images to hide a secret message; then the task should be completed without recording any statistical significance to noise within the carrier. Undetectability is directly impacted by the size of the secret message and its cover image content format (Naoum et al., 2013).

3.2.4 Robustness

The robustness concept is referring to the capability of the embedded data to stay intact whenever the stego-image went under a transformation, this will include the linear and the non-linear filtering; random noise addition, lossy compression, scaling and rotations (Yadav et al., 2011).

3.2.5 Capacity

The capacity concept in data hiding is referring to the total number of bits that are both successfully hidden and then recovered by a steganographic system (Vijayakumar, 2011).

3.3 General Steganography Framework

A general steganography framework is based on the assumption that a sender wishes to send a message to a receiver via the steganographic transmission. Thus, the sender will first start with a cover message which is an input to the stego-system where in which the embedded message will be hidden, and this hidden message is named the embedded message. A steganographic algorithm conjoins the cover message with the secret message which is something going to be hidden in the cover (Jalab et al., 2010).

The algorithm might or might not employ a steganographic key (stego-key), which is an extra secret data that might be required during a hiding process. Usually, the same key will be needed in order to extract the embedded message again. The output of the steganographic algorithm is called stego message. Both of the cover message and the stego message must be of the same data class; however, the embedded message could be of a different data class. And in order to extract the embedded message, the receiver has to invert the embedding process. Below is figure 3.1, which is illustrating general steganography framework (Jalab et al., 2010).

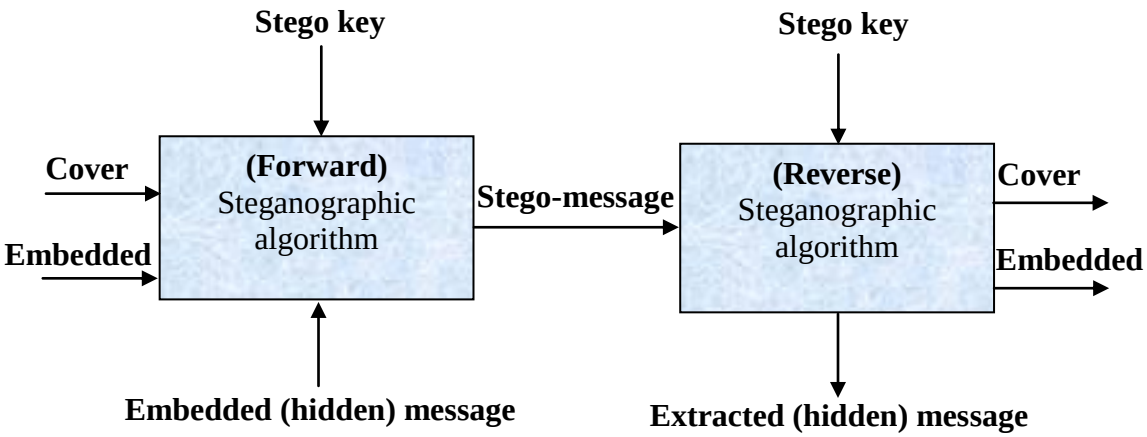


Figure 3.1: General steganography framework, adapted from (Jalab et al., 2010)

3.4 Steganography Methods

Steganography includes different methods for transmitting secret messages in such a way that the entity of the embedded message is unrevealed. Carriers of such type of messages might look like innocent sounding text, audio, images, video, disks, protocols, network traffic, the way circuits or software are arranged, or any others digital media represent code or transmission (Johnson et al., 2001).

3.4.1 Steganography in Text

Steganographic methods can perform a direct encoding for the information in the text. This is because a written text contains less information redundancy which could be employed for a secret communication, in such a manner for profiteering the natural abundance of languages, or it is employed within a text format by modifying the inter-word space or the inter-line space. Furthermore, several ways were suggested in an aim to directly store information in the messages, including replacement of words by synonyms, infrequent typing or spelling errors, commas omitted. However, most of the previously mentioned options are not serious, due to their heavily degradation effects on the text. In addition, the need of the user interaction during an embedding task made it impossible to turn the process into an automated one (Katzenbisser & Petitcolas, 2000).

3.4.2 Steganography in Image

There are numerous and diverse methods for hiding information in images, where it is possible to modify attributes including the contrast, the colors or the luminance, in order to hide secret messages. Suggested methods are capable of hiding data in images with no virtual impact on the human sensory system. Once an image is considered for hiding information in it, then the structure of the image itself should be also considered just like the palette. The most popular method to hide information in an image is the least significant bit (LSB) insertion or manipulation. It is a common and plain tactic for embedding data in a cover, yet such approach is vulnerable to even a slight image distortion. When we want to hide an image in the LSBs of each byte of the 24-bit image, we can store 3 bits in each pixel, and at the end, the resulting stego-image will look congruous to the cover image for the human eye (Pejas & Piegat, 2006).

3.4.3 Steganography in Video

If the information is hidden inside a video, then the program which is hiding the information will usually apply the discrete cosine transform (DCT) method. DCT works by slightly changing each of the images in the video so that it is not noticeable by the human eye. Precisely, DCT usually works by altering values of certain parts of the images, through rounding them up. In summary, it is noticed that a steganography in videos is similar to that steganography in images, aloof from the information that have been hidden in each frame of the video. Furthermore, when only a small amount of information is hidden inside a video, it isn’t generally noticeable at all, however the more information that is hidden the more noticeable it will turn to (Bhattacharyya et al., 2010).

3.4.4 Steganography in Audio

The LSB method is one of the precocious methods that were applied for the digital audio information hiding where each bit from the message is hidden in the least significant bit cover audio in a predefined way, as explained in figure 3.2. The LSB method allows high embedding capacity for data, with a great ease to accomplish the task alone or conjoin with another hiding techniques. However, this technique is also recognized by its low robustness to noise addition, which decreases its security performance since it becomes vulnerable to simpler attacks (Djebbar et al., 2012).

Figure 3.2: LSB in 8 bits per sample signal is overwritten by one bit of the hidden data, adapted from (Djebbar et al., 2012)

3.5 Steganographic Techniques

Numerous approaches were selected in order to classify the steganographic techniques. One of these approaches is by categorizing steganographic techniques based on the applied modifications on a cover during an embedding process. This is discussed in the following sections.

3.5.1 Substitution Systems

The basic substitution systems attempt to encode secret information through substituting insignificant parts of the cover with the secret message bits. A receiver will be able of extracting the information only if he has been informed about the positions where secret information was embedded, and since just minor modifications were made

within the embedding process, secret information will not be observed by a passive attacker (Hmood et al., 2010). This system is consisting of various techniques which will be discussed in detail within the following subsections.

A. Least Significant Bit (LSB) Substitution

An embedding process will first include the selection of a cover elements subset $\{j_1 \ldots\ldots J_{l\ (m)}\}$ and then performing the substitution operation $\boldsymbol{c_{ji} \leftrightarrow m_i}$ on them, which also involve the LSB exchange of $\boldsymbol{c_{ji}}$ by $\boldsymbol{m_i}$ ($\boldsymbol{m_i}$ can be either 1 or 0). During an extraction process, the LSB of a selected cover-element is extracted and lined up in a way to recover the secret message (Katzenbisser & Petitcolas, 2000).

For instance, a digital image is consisting of a color matrix and intensity values. in a typical gray scale image, an 8 bits/pixel are employed, while in a typical full-color image, there are 24 bits/pixel, where 8 bits are appointed to every color ingredient. The simplest steganographic techniques directly embedding the bits of a message into the least-significant bit plane of the cover image in a deterministic sequence. The Modulation of the least-significant bit does not produce a difference in a human perceptibility, as the capacity of the total alteration is considered slight. The LSB embedding advantages are characterized by its simplicity and high perceptual transparency, thus, many techniques employ these methods. In contrast, there are many disadvantages regarding the robustness, tamper resistance, and other security issues. It is worth mentioning that LSB encoding is strictly sensitive to any sort of filtering or manipulating processes regarding a stego-image (Mathkour et al., 2009).

B. Pseudorandom Permutation

This technique was particularly initiated as a one solution for the defects of the previously discussed method. Each image's sender and receiver have a specific password and a designated stego-key which are employed like a seed for the purpose of generating a pseudo-random number. This also creates a sequence like $\{X_1, X_2 \ldots X_l (m)\}$ which is basically used as an index to get an access to the image pixel. The bit of a message m_l is embedded in a pixel C_{Xl} of a cover image, where the index X_l is represented by the pseudo-random number generator (Rodrigues et al., 2004).

It is compulsory that all the methods which are based on the pseudo-random number generator must employ an array to control the collisions. The main two characteristic features of the pseudo-random permutation methods are the use of a password to gain an access to the message, as well as the well-spread message bits over an image (Rodrigues et al., 2004).

3.5.2 Transform Domain Techniques

It has been proven that the substitution modification techniques are considered as plain ways to embed information, yet they are highly susceptible to even a minor modification. An attacker can frugally apply the signal processing techniques in order to devastate secret information. In several reported cases, even the tiny changes can lead to a loss in compression system and accordingly to a total information loss as well. During the earlier times of the steganography systems development, it was found out that embedding information in the frequency domain of a signal can allow more robustness than embedding rules operating in the time domain. Today, most of the known robust steganography systems are actually operating within some sort of a transform domain (Zaidan et al., 2009).

Transform domain methods are hiding messages in significant areas of the cover image in order to make them more robust to attack, including cropping, compression, and adding noise as well as some image processing. Even that these techniques are more robust to several sorts of signal processing, yet they are remaining imperceptible to the human sensory system. In fact, various transform domains do exist, one method is employing the discrete cosine transformation (DCT) as a vehicle to embed information in an image, while another method is employing the wavelet transforms. Practically, during embedding process, these transforms embed a secret message by modifying the transform coefficients of the cover message (Hmood et al., 2010).

A. Wavelet Transform

The wavelet transform is involved in converting a spatial domain image into a frequency domain, as it can provide the representation of a time-frequency. The wavelet transform is initiated by the repeated filtering of the image coefficients on a row-by-row and a column-by-column basis. Substantially, the utility of wavelets in an image steganography lies in the fact that the wavelet transform fairly separates high-frequency and low-frequency information on a pixel-by-pixel basis. Once a cover image is passed through a wavelet filter bank, the image will be convolved with a wavelet low pass filter, granting smoother versions of the initial input image or it will be convolved with a high pass filter, resulting in a final detailed band. Such decomposition can be carried up to log2 (min (height, width)). On the other hand, final level decomposition for an image low pass coefficients constitutes an approximation band (George, 2012).

3.6 Discrete Wavelet Transform

The idea on which the discrete wavelet transforms (DWT) performance based is that one dimensional signal will be divided into two parts, one is a high frequency part and another is a low frequency part. Later, the low frequency part will split into two additional parts and the analogous process will go on until reaching the desired level. As for the high frequency part of the signal, it is contained by the signal's edge components. In each level of the DWT decomposition, an image will separate into four other parts which are referred to as the approximation image (LL), in addition to the horizontal (HL), the vertical (LH) and the diagonal (HH) for a detailed components. Precisely within the DWT decomposition, an input signal must be the multiple of 2^{n}. Where *n* represent the number of levels. Furthermore, and in order to analyse and syntheses the original signal, a DWT is capable of providing all the requested information with a less computation time (Rahman, 2013).

3.6.1 Haar- Discrete Wavelet Transform

Haar wavelet is classified as the simplest and most commonly employed wavelet. It can perform in two different ways; first is the horizontal way and second is the vertical way. Haar wavelet functions by scanning the pixels from left to right in a horizontal direction, next it will perform an addition and subtraction operation on the neighboring pixels which multiplied by a scaling function for Haar wavelet is $1/\sqrt{2}$. At last, the final result of the addition on left half and the addition on the right half will be stored, and by considering the starting pixel as **A** and the neighboring pixel as **B** it will be elucidated in the following formulas (Mahajan & Kranthi, 2014).

$$\textbf{Sum on left side} \quad = \frac{\mathbf{A+B}}{\sqrt{\mathbf{2}}} \tag{3.1}$$

$$\textbf{Difference on right side} \quad = \frac{\mathbf{A-B}}{\sqrt{\mathbf{2}}} \tag{3.2}$$

The process should be repeated until it can cover all the rows, while the pixel sum will be represented by a low frequency, and the difference is represented by a high frequency. After accomplishing the previously described steps, it is possible to scan the pixels from the top to the bottom in a vertical direction. At the end, the addition and subtraction operation will be multiplied by $1/\sqrt{2}$, and the result of the addition on the top and the subtraction on the bottom will be stored. Such operations analysis and synthesis are preformed through filter bank (Mahajan & Kranthi, 2014).

3.6.2 Two-Dimensional Haar- Discrete Wavelet Transform

Fundamentally, the two dimensional discrete wavelet transform is a one dimensional analysis for a two dimensional signal, which means that it is operating only on a one dimension at a one time, by analyzing the rows and the columns of an image in a separable style. At first an analysis filter will be applied to the image rows producing another two new images, where one image is set of coarse row coefficients, and the other one is set of detailed row coefficients. Next phase is applying analysis filter to the columns of each new image producing four different images named as sub bands. Rows and columns are analyzed with a high pass filter that is designated with an **H**. In the same way, rows and columns are analyzed with a low pass filter that is designated with an **L**. For instance, if a sub band image was produced by employing a high pass filter on the rows and a low pass filter on the columns, thus, it is called the **HL** sub band. figure 3.3, elucidates this entire process (Mistry & Banerjee, 2013).

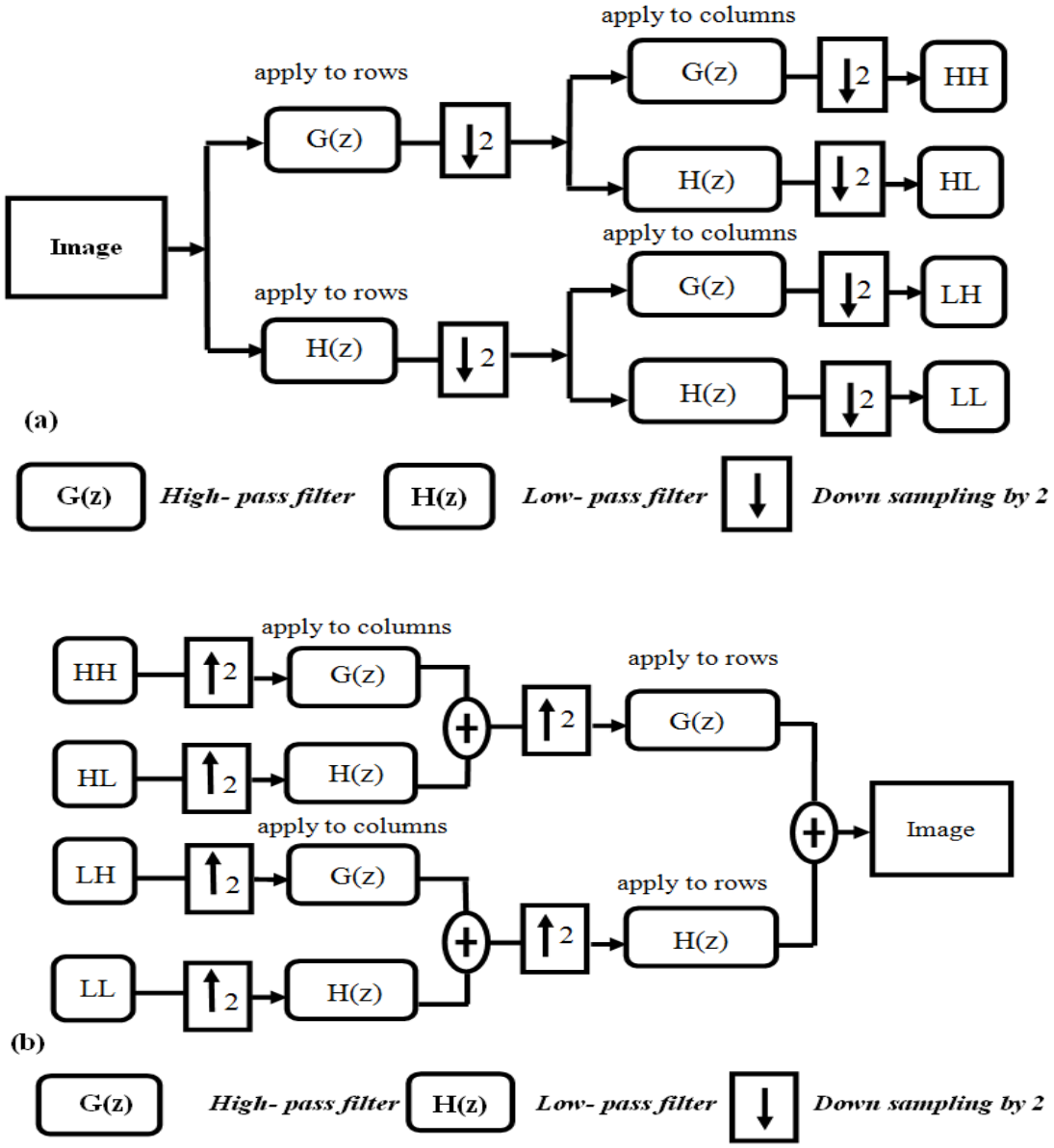
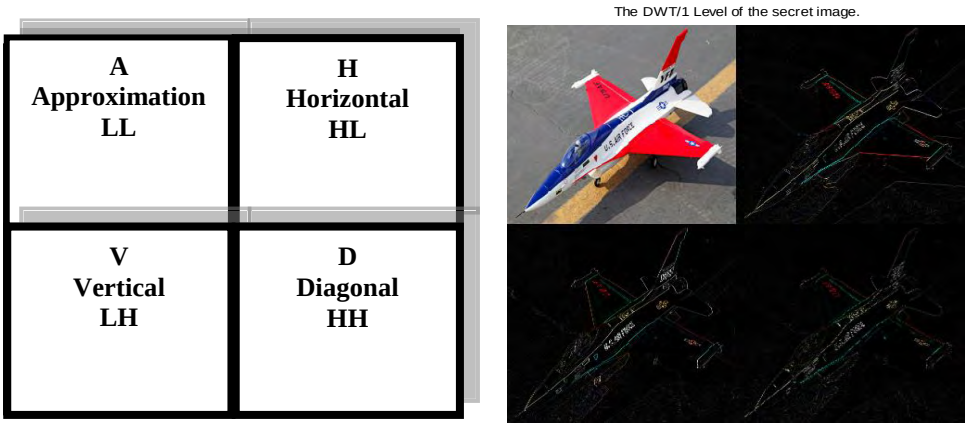


Figure 3.3: 1- Level 2D –DWT (a) Decomposition, (b) Reconstruction,
(Mistry & Banerjee, 2013)

Figure 3.3 a, is illustrating the 1-Level 2D-DWT image decomposition, where the operation starts by applying the one-dimensional DWT, along the rows of the image, then the results are decomposed along the columns. The final operation results are represented in four decomposed sub band images which are referred to a low-low (LL), a low-high (LH), a high-low (HL), and a high-high (HH) frequency bands as shown in figure 3.4.



(a) Decomposition structure **(b) Decomposition image**

Figure 3.4: 1-Level 2D- DWT decomposition of an image

3.7 Steganography Attackers

Generally, the process of breaking any steganography system needs three major steps, *detecting*, *extracting*, and *information displaying*. In case the attacker was able to prove that a secret message exists, then the entire system will be unsecured. The next paragraph will discuss the main three types of the attackers: *Passive*, *Active* and *Malicious* attacker.

3.7.1 Passive Attacker

Passive attacker act by monitoring the communications without any sort of interference, thus, if an attacker is being restricted from modifying the stego-files contents during the communication process then it is called a *passive attacke*r. The passive attack functions by either preventing or permitting the message delivery, and accordingly, the communication between two parties will be blocked once the attacker suspected the presence of a secret communication, else the communication will continue relaying (Cox et al., 2007).

3.7.2 Active Attacker

Active attackers are not capable of extracting or proving the existence of a secret message, thus, they can simply add a random noise to the transmitted cover in an aim of destroying the information. In the case of digital images, an attacker can either apply image processing techniques or convert the image to another file format, consequently, all of these techniques can be mischievous to the secret communication (Katzenbisser & Petitcolas, 2000).

3.7.3 Malicious Attacker

The malicious attacker turned the robustness factor into inadequate one, particularly in cases where the embedding methods are independent from a part of secret information which are shared by the sender and the receiver. Then, an attacker will be able to forge a message as long as the recipient is incapable of verifying the validity of the sender's identify. In order to avoid such attacker, the applied algorithm should be robust as well as secure (Katzenbisser & Petitcolas, 2000).

3.8 Countermeasures Against Attacks

The prime purpose of a countermeasure is to prevent successful attacks. It comes in two possible ways:

- Countermeasures to deter the detection which might include data hiding in perceptually less significant areas within a cover or scattering the message throughout a cover (Johnson et al., 2001).
- Countermeasures deter the distortion which might include embedded data in perceptually more significant parts of a carrier, in an aim to make either the distortion or the removal of the embedded information more complicated.

Despite the previously mentioned facts, countermeasures are not always there for all kinds of attacks particularly if the embedded information is completely overwritten or destroyed, at such case, countermeasures are helpless in recovering the embedded information (Johnson et al., 2001).

3.9 Fidelity Criteria

The fidelity criteria are classified into two classes; the *objective fidelity criteria* and *subjective fidelity criteria*. The *objective fidelity* criteria is adapted from the digital signal processing as well as the information theory, in order to grant us the ability to provide specific equations that can be employed to measure the error amount within a processed image in comparison to a known image, and in such case, we will refer to the processed image as a reconstructed image. On the other hand, the *subjective fidelity* criteria will first require the definition of a qualitative scale in order to start assessing an image quality, and later, the resulted scale can be employed by a human test subject to determine an image fidelity (Umbaugh, 2011).

In particular, the objective measures are categorized into two types; *the peak signal-to-noise ratio (PSNR)* and *the mean square error (MES).* These two types are discussed in the following paragraph.

3.9.1 Peak Signal -to- Noise Ratio (PSNR)

Practically, one of the most commonly used objective methods is the PSNR metric, specifically within the domains of watermarking and steganography. After accomplishing either the watermark or the information embedding, high PSNR values of the stego image will coincide to a high similarity versus the original image (Yan & Weir, 2010). Accordingly, the PSNR is defined by a formula as follows:

$$\text{PSNR} = 10\log_{10} \frac{(255)^2}{\frac{1}{(N \times M)} \times \sum_{r=0}^{M-1} \sum_{c=0}^{N-1} [I_1(r,c) - I_2(r,c)]^2} \quad (3.3)$$

Where:

$I_1(r,c)$: The intensity value of the pixel in a cover image.

$I_2(r,c)$: The intensity value of the pixel in a stego image.

M×N : The size of an image.

r and c : The number of rows and columns.

3.9.2 Mean Square Error (MSE)

The mean square error is capable of measuring the statistical difference of the pixel values between the original and the reconstructed image. In fact, the MSE represents the commutative square error between the original image and the stego image, where a lower MSE value indicates a better image quality as well as lesser distortion within the stego image (Kamau et al., 2012). Mathematically, the MSE is defined by the following formula:

$$\text{MSE} = \frac{1}{M \times N} \sum_{r=0}^{M-1} \sum_{c=0}^{N-1} \left[I_1(r,c) - I_2(r,c) \right]^2 \quad (3.4)$$

$I_1(r,c)$: The intensity value of the pixel in a cover image.

$I_2(r,c)$: The intensity value of the pixel in a stego image.

M×N: The size of an image.

r and c : The number of rows and columns.

Chapter Four

Artificial Neural Networks

4.1 Introduction

An artificial neural network (ANN) is representing a model of information processing which initially employs the mimicking of the human biological nervous systems, such as the way human brain processes information. The principal element of such pattern is the novel structure in which the information is processed. It consists of a huge number of highly interconnected processing elements symbolized by what so called neurons which are working in harmony and unity in order to solve specific assigned problems. An ANN system is acting just like people, which means that it learns by sitting various examples, in order to be configured for a later specific application, including pattern recognition and/or data classification, through a learning process. Learning within biological systems involves adjustments to what so called the synaptic connections which exist between the neurons, and the same thing also applies to the ANN_s systems (Choudhury et al., 2007).

The Artificial Neural Networks has numerous types of applications. Since the first neural model proposed by McCulloch and Pitts (1943), a huge number of different models have been proposed and developed for ANN. These models might be different in terms of their transfer functions, network topology, the format of accepted values and the algorithms used in learning process. Moreover, there are variety of hybrid models where each neuron has more properties than that own by pure models. Our proposed system apply the neural network model that adapt the resilient back propagation algorithm for learning, the weights where this model is considered one of the most

common models that used in the ANN_S. The main function of ANN is to process the information, so it used in many fields related to it. ANNs are used in various engineering applications including pattern recognition, data compression and forecasting. In addition, there are a broad variety of ANNs that used to model the real biological neural networks, and studying and control animals and machine behaviors (Jayasimman & George, 2013).

4.2 The Neural Network Mathematical Model

The artificial neuron model which is the widely used one in artificial neural networks is represented in figure 4.1, with slight minor modifications. A single artificial neuron is the basic element of the neural network. Precisely, the artificial neuron has an *N* input that is denoted as u_1,u_2, ...u_N. Every line which is connecting these inputs to a neuron is assigned as a *weight*, and are denoted as w_1, w_2,..,w_N respectively.

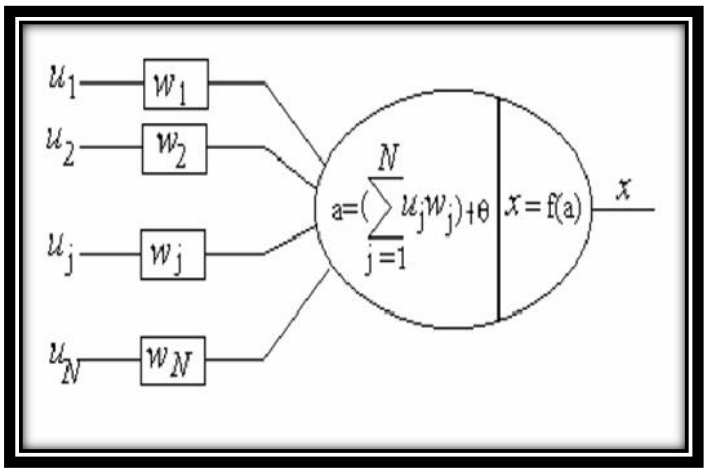


Figure 4.1: The neural network mathematical model, adapted from (Odabas et al., 2013)

Usually within an artificial model system, weights will correspond to the synaptic connections by the same way within the biological neurons. Furthermore, the *threshold* in artificial neuron is usually represented by θ, where the activation corresponding to the graded potential is expressed by the following formula (Odabas et al., 2013).

$$a = \sum_{j=1}^{N} w_j u_{j+\theta} \tag{4.1}$$

The inputs and the weights are considered as real values, where a negative value for a weight indicates an inhibitory connection, while a positive value indicates an excitatory one. Despite the fact that biological neurons have a negative value, yet it might be assigned as a positive value within an artificial neuron model. If θ is positive, then it is usually referred to it as a bias. For the mathematical suitability, we will use a (+) sign in the activation formula. Occasionally, and for simplicity purposes, the threshold is combined to the summation part through assuming an imaginary input equals to u_0 = +1 and a connection weight equals to w_0 = θ. And accordingly, the activation formula becomes (Odabas et al., 2013):

$$a = \sum_{j=1}^{N} w_j u_j \tag{4.2}$$

The output value of the neuron is representing the function of its activation which analogous to the firing frequency of the biological system neurons:

$$x = f(a) \tag{4.3}$$

4.3 Architecture of Neural Network

From a different point view, ANNs is counted as a direct method of weighting graphs, where artificial neurons nodes and directed edges (with weights) are acting as connections between the neuron outputs and its inputs. As figure 4.2 depicts, the First layer of neural network is the input layer which contains *(n)* neurons whereas the last layer is the output layer that contains *(m*) neurons. Input to neuron **x**=(x_1,x_2,x_3…x_n) is a feature vector in n-dimensional feature space.

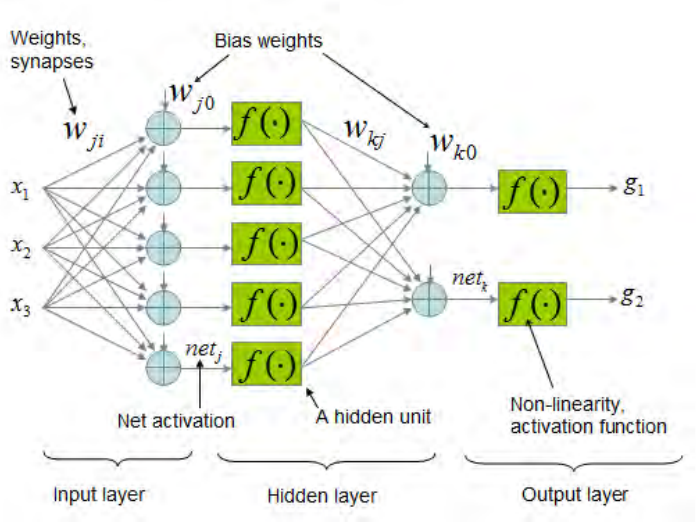


Figure 4.2: Architecture of neural network

The activity of input units represents the basic raw information which is fed into the network, while the activity of every hidden unit is determined by the input units' activities as well as the weights on the connections between the input and those hidden units. The behavior of the output units depends from one side on the activity of the hidden units, and from other side on the weights between the hidden and output units, so reaching an output layer neuron, value from every hidden layer neuron is multiplied by a weight (w_{kj}), which resulted in weighted values that will be added producing a combined value net. The weighted sum (net_j) is fed to a transfer function (logistic function) *f(.)*, which outputs (g_1 and g_2). The g's values are network outputs.

In our thesis, we applied the following modified sigmoid transfer function which is represented in the following formula:

$$f(x) = \frac{1 - e^{-ax}}{1 + e^{-ax}} \qquad \text{(Scalero \& Tepedelenlioglu, 1992)} \qquad (4.4)$$

Naoum (2011) elucidated that for a sigmoid (logical) activation function, the output continuously keep varying in a non-linear relation as long as the input changes. The sigmoid function have an output which is bounded with a lower limit bound

(0 or -1) and an upper limit bound (+1). The sigmoid function is differentiable real function and owns a positive derivative as it shown in figure 4.3.

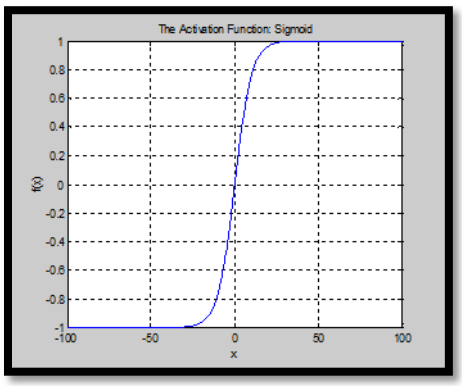


Figure 4.3: Sigmoid transfer function (Scalero & Tepedelenlioglu, 1992).

4.4 The Learning Process

We can categorize the learning situations in two distinct sorts. These are:

4.4.1 Supervised Learning

The supervised learning is the kind of learning that needs to incorporate an external teacher, so that each output unit separately told what exactly should be the desired response for the input signals. During the learning process, a kind of global information might be required. Patterns of supervised learning can include the error-correction learning, the reinforcement learning and the stochastic learning (Sagar et al., 2011). In our thesis we made use of the supervised learning as shown in figure 4.4.

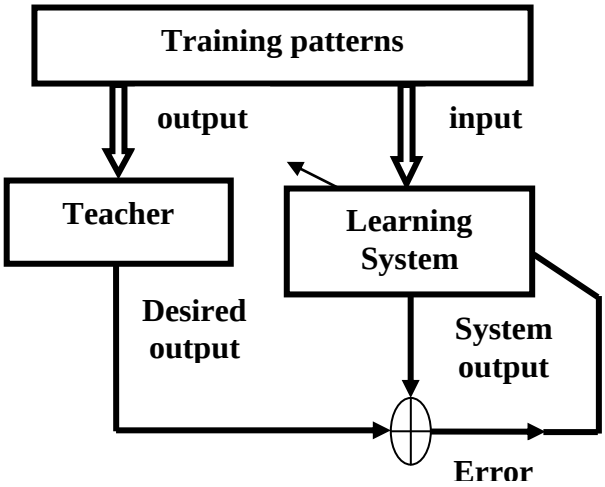


Figure 4.4: Overview of supervised learning (Chow & Cho, 2007).

4.4.2 Unsupervised Learning or Self- organization

Unlike the supervised learning, the unsupervised learning does not own a teacher within a training data set, instead, the learning process of the unsupervised neural networks has took place by a self-organizing behavior. During the course of training, no external factor is employed to affect the weights adjustment of the network, and the correct outputs are not available during the course of training. Generally, a typical unsupervised network consists of an input layer, and a competitive layer. The neurons on the competitive layer keep competing with each other via a simple competitive learning rule in order to represent a given input pattern in a best way. Additionally, through the competitive learning the network output automatically reflects some statistical characteristics of input data including data cluster and topological ordering (Chow & Cho, 2007).

The unsupervised learning objectives are to find a specific kind of regularity manner within the data represented by the exemplars. The self-organizing map (SOM) is the most widely used unsupervised neural networks. Below is figure 4.5 which represents the unsupervised learning process (Chow & Cho, 2007).

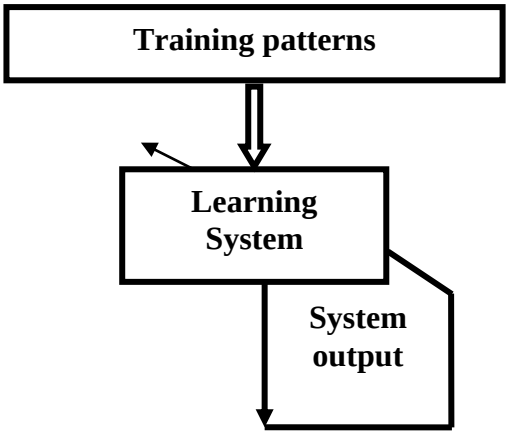


Figure 4.5: The unsupervised learning process (Chow & Cho, 2007).

The self-organizing map (SOM) network is originally designed to solve problems which involve tasks like clustering and visualization; it can be successfully used as a classification tool. The self-organizing map (SOM) network is a special type of neural network that can learn from both complex and multi-dimensional data, and it is also capable of transforming these data into visually decipherable clusters. The main function of SOM networks is to map the input data from an n-dimensional space into a lower dimension (usually one or two-dimension). SOM employed a various approach which differ from other approaches that are used by the another neural network where the SOM network used unsupervised training where during such learning process, the processing neurons in the network adjust their weights based on the lateral feedback connection (Kiang, 2001).

4.5 Architectures for Training of an Artificial Neural Network

The main character which is of a great significance for a neural network is its ability to learn from its surrounding environment, and later upgrade its performance through that learning process. The improvement in performance takes place over a period of time in accordance to certain prescribed measurements. A neural network learns from its environment through an interactive process of adjustments that is applied on its synaptic weights as well as bias levels. Thus, after every learning process refinement the network will end to become a more knowledgeable system about its environment (Haykin, 1999).

The best-known example of a neural network training algorithm is back propagation where it still has advantages in some circumstances, and it is the easiest algorithm to understand. The following subsections explain the back-propagation algorithm and the adaptive back propagation algorithm that we adopted in our thesis.

4.5.1 Back-propagation Neural Network

It is very well known that the backpropagation algorithm is a quite useful type of algorithm for training of neural networks. However, the main difference in the backpropagation algorithm is presenting the neural network with training data, where each item of the training data is given to the neural network, while the error is calculated between the actual and the expected outputs of the neural network. After that, the weights and threshold are modified for a greater chance of network to renaissance the correct result when this network is also presented in the forward layer with the same input (Heaton, 2008).

The backbone of the adaptive resilient back-propagation training algorithm that will be explained in the following subsection is the back-propagation algorithm that proposed by (Lippmann, 1987).

Step 1: Initialize Weights and Offsets. Sets all weights and node offsets to small random values.

Step 2: Present Input and Desired Outputs.
Present a continuous valued input vector x_0, $x_1,\ldots$ xn_{-1} and specify the desired outputs $d_0,d_1,\ldots,d_{m-1}$. If the net is used as a classifier then all desired outputs are typically set to zero except for that corresponding to the class the input is from. The input could be new on each trial or samples from a training set could be presented cyclically until weights stabilize.

Step 3: Calculate Actual Outputs. Use the sigmoid nonlinearity from (4.5) formula to calculate the outputs y_0, $y_1,\ldots y_{m-1}$. In our thesis we used modified sigmoid logistic non-linearity function as an activation function.

$$f(x) = \frac{1 - e^{-ax}}{1 + e^{-ax}} \tag{4.5}$$

Step 4: Adapt weights. Use a recursive algorithm starting at the output nodes and working back to the first hidden layer. Adjust weights by:

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} + \mu\delta_j x'_i \tag{4.6}$$

where $w_{ij}^{(t)}$ is the weight from hidden node i or from an input to node j at time t, x'_i is either the output of node i or is an input, μ is the gain term, and $\delta_j$is an error term for node j. If node j is an output node, then

$$\delta_j = y_j\,(1 - y_j)(d_j - y_j)\,, \tag{4.7}$$

Where d_j is the desired output of node j and y_j is the actual output.
If node j is an internal hidden node, then

$$\delta_j = x_j\,(1 - x'_j)\sum_k \delta_k w_{jk} \tag{4.8}$$

Where k is over all nodes in the layers above node j. Internal mode thresholds are adapted in a similar manner by assuming they are connection weights on links from auxiliary constant-valued inputs.

Step 5: If the Mean Square Error is above some predefined value then repeat by going to step 2. Otherwise go to step 6.

Step 6: Stop and store the gained optimal weights .

4.5.2 Adaptive Back-propagation Learning: The (RPROP) Algorithm

In general the main goal of using the back-propagation learning algorithm is to minimize the error function $E(w)$ where this function is depended on the weight vector in each layer, then by taking the partial derivative for each weight $\frac{\partial E}{\partial w_{ij}}$ and performing a simple gradient descent, the error function minimizing is achieved as the following equation shows:

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} - \epsilon\frac{\partial E}{\partial w_{ij}}(t) \tag{4.9}$$

Where w_{ij} is the weight from neuron i to neuron j. The parameter Є (learning rate) scales the partial derivative of error function $\frac{\partial E}{\partial w_{ij}}$ this parameter affect the convergence time of network. In such a way, if Є chosen to be too small, then the network will need many steps until acceptable convergence is reached, on the other hand, if Є chosen to be too large then the network undergoes an oscillation around convergence which prevent the error to fall below a certain threshold (Riedmiller & Braun, 1993).

To solve this problem another parameter is proposed, as explained in the following equation:

$$\Delta w_{ij}(t) = -\epsilon \frac{\partial E}{\partial w_{ij}}(t) + \mu\, \Delta w_{ij}(t-1) \qquad (4.10)$$

Where μ is the momentum parameter that scales the influence of the previous step *(t-1)* on the current one *(t)*. However, both of ϵ and μ are proved experimentally to be problem-dependent. Thus, adding ϵ and μ parameters to the learning rule has no general overall improvement.

In aim of solving above problems, (Riedmiller & Braun, 1993) proposed a novel technique to change the size of the weight-update $\Delta w_{ij}(t)$ directly by assigning its values and without using the size of error function partial derivative $\frac{\partial E}{\partial w_{ij}}$. The term "RPROP" is standing for 'resilient propagation' and is an efficient new learning scheme which performs a direct adaptation of the weight step based on the local gradient information. The size of the weight-update $\Delta w_{ij}(t)$ follow the following simple rule:

$$\Delta_{ij}^{(t)} = \begin{cases} \eta^{+} * \Delta_{ij}^{(t-1)} & ,if \quad \frac{\partial E}{\partial w_{ij}}^{(t-1)} * \frac{\partial E}{\partial w_{ij}}^{(t)} > 0 \\ \eta^{-} * \Delta_{ij}^{(t-1)} & ,if \quad \frac{\partial E}{\partial w_{ij}}^{(t-1)} * \frac{\partial E}{\partial w_{ij}}^{(t)} < 0 \\ \Delta_{ij}^{(t-1)} & , \quad else \end{cases} \qquad (4.11)$$

$$where\ 0 < \eta^{-} < 1 < \eta^{+}$$

The proposed adaptation rule above works in the following manner: Every time the partial derivative of corresponding weight w_{ij} changes its sign, which indicates that the last weight-update $\Delta w_{ij}(t)$ was too big and the learning algorithm has jumped over a local minimum, the update value Δ_{ij} is decreased by a factor η^-. on the other hand, if the derivative keep its sign then the update value Δ_{ij} is increased slightly to accelerate the network convergence in shallow regions by a factor η^+ (Riedmiller & Braun, 1993).

Once the update value Δ_{ij} for each weight is evaluated, then the weight-update itself follow the following rule :

$$\Delta w_{ij}^{(t)} = \begin{cases} -\Delta_{ij}^{(t)} & ,if \quad \frac{\partial E}{\partial w_{ij}}^{(t)} > 0 \\ +\Delta_{ij}^{(t)} & ,if \quad \frac{\partial E}{\partial w_{ij}}^{(t)} < 0 \\ 0 & , \quad else \end{cases} \tag{4.12}$$

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} + \Delta w_{ij}^{(t)} \tag{4.13}$$

As equations (4.12) and (4.13) show, if the partial derivative of error function is positive which means error increasing, then the weight-update will be decreased by its update-value, and if the partial derivative is negative, then the update-value is added. As an exception case, if the partial derivative of error function changes sign (i.e., the local minimum was missed by too large previous step), then the previous weight-update is reverted (Riedmiller & Braun, 1993).

$$\Delta w_{ij}^{(t)} = -\Delta w_{ij}^{(t-1)}, \qquad if \frac{\partial E}{\partial w_{ij}}^{(t-1)} * \frac{\partial E}{\partial w_{ij}}^{(t)} < 0 \tag{4.14}$$

The pseudo-code that represent the above equations is illustrated in chapter five and implemented using MATLAB language.

Chapter Five

Proposed Artificial Neural Network- Steganography System Implementation

5.1 Introduction

The main problem of image hiding in another host image is the large amount of data that requires a special data embedding technique to obtain enough capacity, transparency and robustness.

Our proposed Steganography system, which embeds (RGB) secret image inside (RGB) cover image chosen by an enhanced resilient back propagation neural network (ERPROP). Our proposed system applies a discrete wavelet transform (DWT) in combination with enhanced resilient back propagation neural network algorithm in the embedding process to achieve a robust and multilayer security system with high invisibility.

In the first algorithm, we use the enhanced resilient back propagation neural network to select the best cover image that will be used to embed the secret image and to approximate the best embedding threshold value for each color layer.

In the second algorithm, we use Haar based-DWT for the cover and the secret image, where the cover image and secret image will be decomposed into four and one level DWT respectively. Each level of decomposition produces four sub bands of coefficients, low pass sub band (LL), and three other corresponding to Horizontal (HL), Vertical (LH), and Diagonal (HH) high pass sub bands. The coefficients of these sub bands will be used to embed the secret image into cover image.

The proposed system consists of two main phases: the embedding phase and the extraction phase. In the embedding phase, the combination of (ERPROP) and (DWT) algorithms takes the secret image and the cover image as inputs, and the stego image will be created, whereas, in the extraction phase, the stego image will be decomposed to extract the secret image once again.

5.2 Pre-Embedding Stages:

The embedding phase is proceeded by three main stages: secret image selection and processing stage, best cover image selection and processing stage and best embedding threshold selection stage.

5.2.1 Secret Image Selection and Processing Stage

In this stage, the secret image is chosen manually, and then processed to get the secret sub bands bit streams. Then, the secret sub bands bit streams are encrypted by key bit streams generated using Fibonacci Linear Feedback Shift Register (FLFSR). The encrypted sub band bit streams will be embedded in the cover image. As shown in figure 5.1 below.

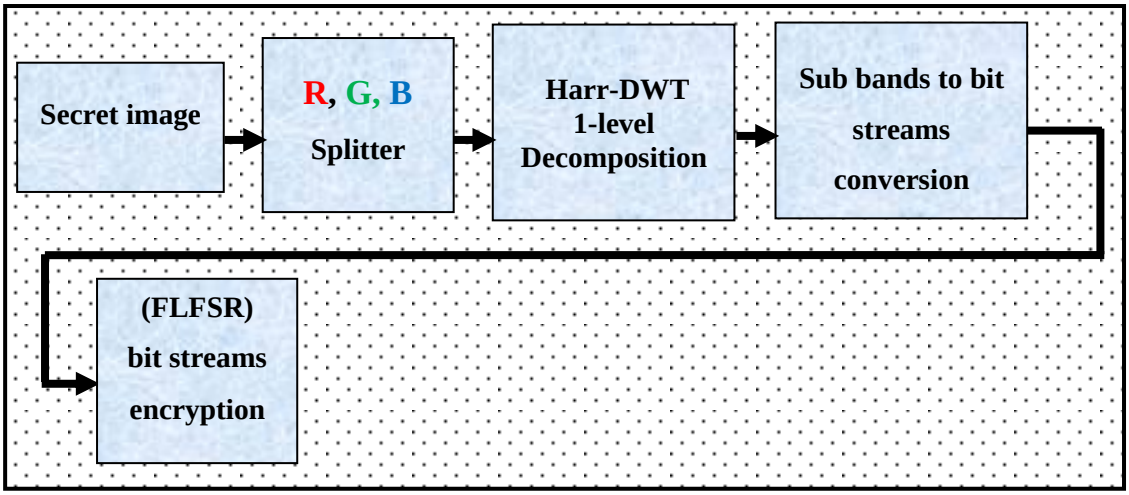


Figure 5.1: Secret image selection and processing stage block diagram

A. Secret Image (RED, Green, Blue) Splitting

The first step of secret image processing is to separate it into (Red, Green ,Blue) color layers, as shown in figure 5.2.

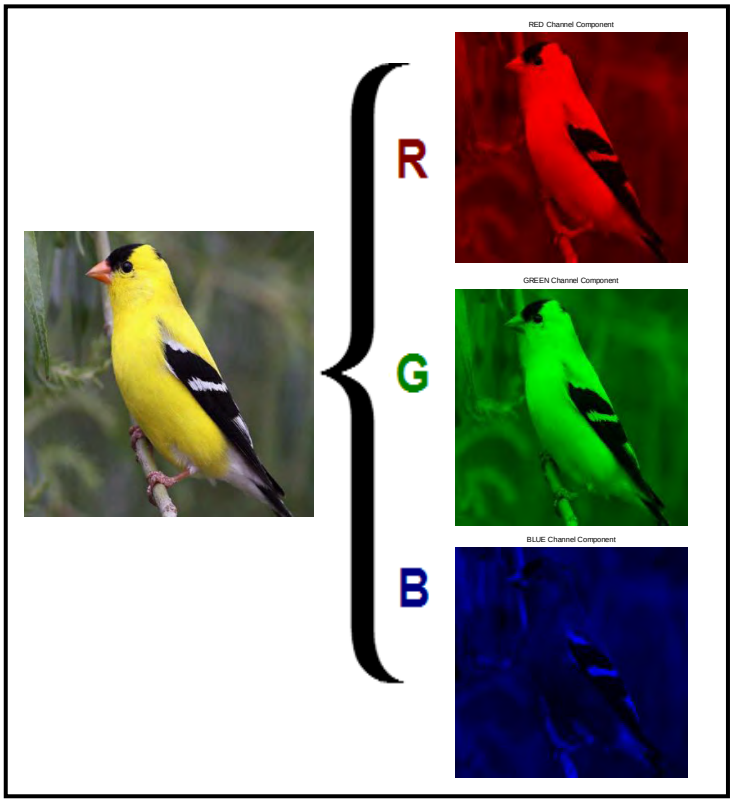


Figure 5.2: RGB layers separation of secret image

B. Discrete Wavelet Decomposition of Secret Image

Discrete wavelet transform of first level is applied to each color layer of secret image, where each color layer of secret image is decomposed to its sub bands (Approximate, Horizontal, Diagonal and Vertical) as shown in figure 5.3 and figure 5.4 respectively.

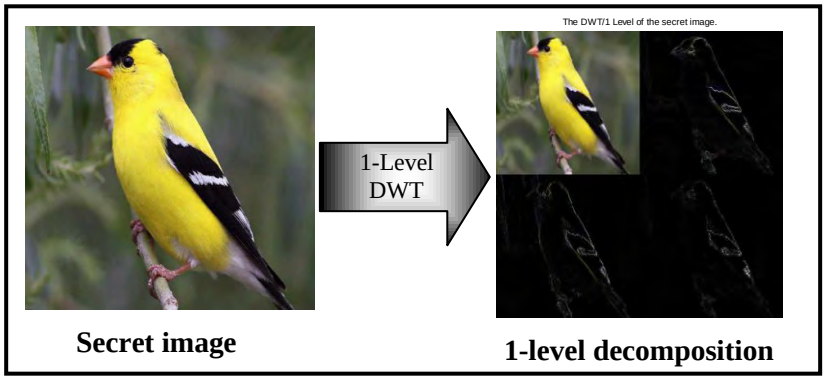


Figure 5.3: 1-level decomposition of the full color (RGB) secret image

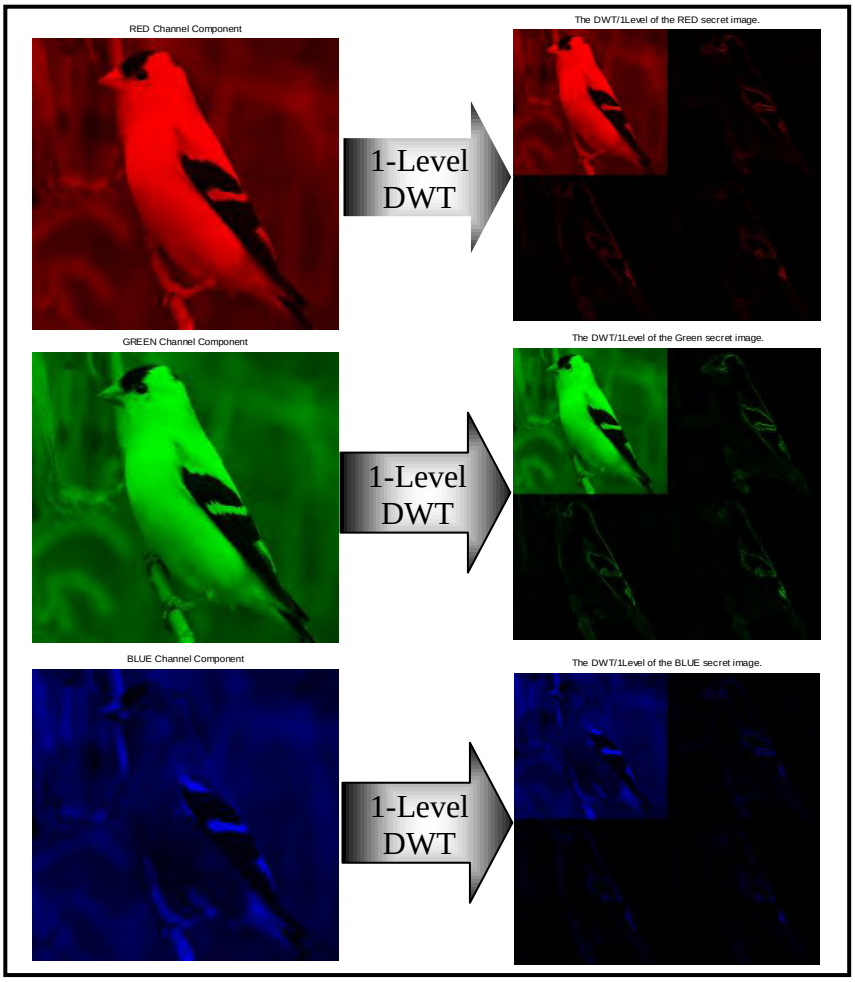


Figure 5.4: 1-level decomposition of Red, Green and Blue layers of secret image

C. Conversion of Secret Image Sub bands to Bit Streams

After the DWT is applied to each color layer of secret image then each color layer will be processed separately. Color layer sub bands will be converted to bit streams, and yield four bit-streams (bit stream for each sub band), where each coefficient is transformed into 16 bits and the bits of all coefficients of a sub band are concatenated to compose the whole bit stream. Figure 5.5 demonstrates the bit stream conversion of sub bands coefficients of the red layer and this approach applied for the rest layers. As another case study, it is possible to convert the coefficients to 24 bits instead of 16-bits and it gives better PSNR values, however it consumes much time and computational power especially in extraction process.

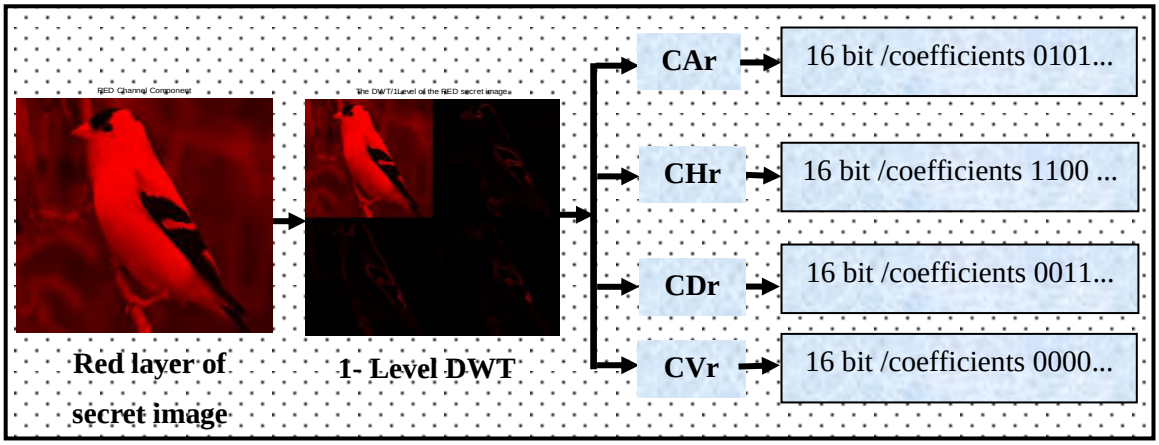


Figure 5.5: Bit stream conversion of secret image coefficient of red layer

D. Key Generation and bit Streams Encryption

Once the bit stream of each sub band is available, the encryption step begins. The encryption will cipher the bit stream using key produced by modified Fibonacci Linear Feedback Shift Register (FLFSR) (Goresky & Klapper, 2002).

Modified Fibonacci Linear Feedback Shift Register consists of one linear feedback shift register (LFSR) whose length, feedback function, and output function are shown in the following figure 5.6.

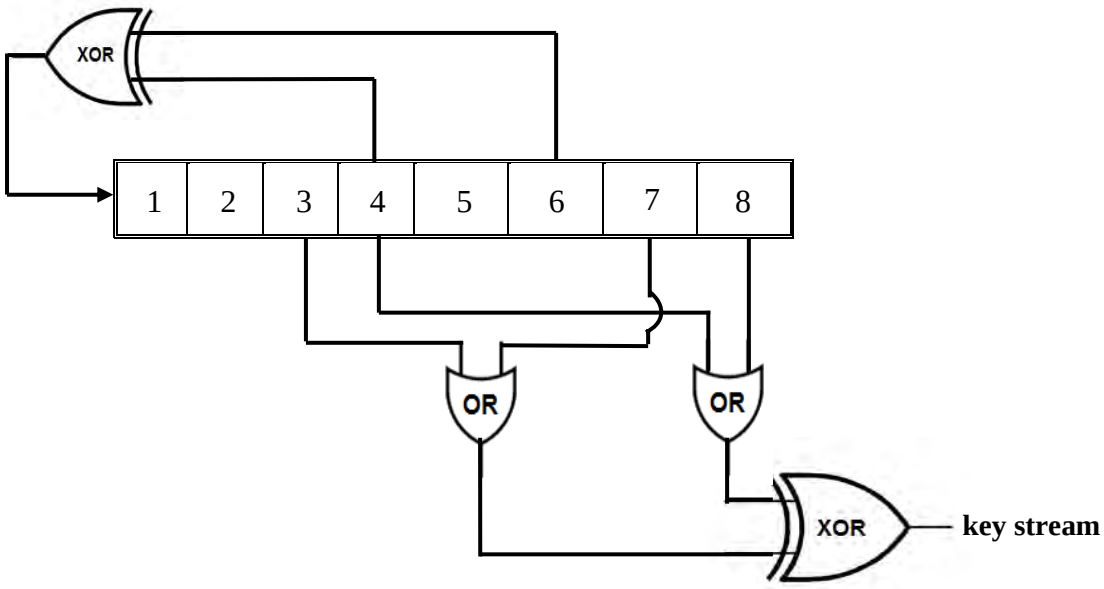


Figure 5.6: A 8-bit modified fibonacci linear feedback shift register

Referring to figure 5.6, Modified (FLFSR) architecture consists of a linear feedback register with an (XOR) gate on its fourth and sixth bit then fed back to the first bit each time it's shifted from left to right. The other tabs of register are processed in the output function as follows:-

The third tab and the seventh tab share the operation of (OR).

The fourth tab and the eighth tab share the operation of (OR).

Results of the above (OR) operations share (XOR) operation.

Initial state of the register is set using the first pixel of each color layer, and this 8-bits pixel is considered the seed value of the register, where we used a separate seed for each color layer, so we have a separate encryption key for each color layer and this increases the security level of our proposed system.

Then the operation of shifting each time produces new bit key. We generate key bits as much as we need to form a stream of ciphering key matches in length of sub band bit stream. Then we (XOR) both the bit key with sub band bit stream together to have our encrypted sub band bit stream.

5.2.2 Best Cover Image Selection and Processing Stage

In order to achieve high PSNR values in the process of embedding the secret image inside the best cover image, a hybrid system built upon two different types of artificial neural networks will be used to select the best cover image among a set of cover images. The first network is self-organizing map (SOM) neural network, which is an unsupervised artificial neural network, and the second one is the enhanced resilient back propagation neural network. Figure 5.7 illustrates the major building blocks of this stage and the function of each block is elaborated as follows.

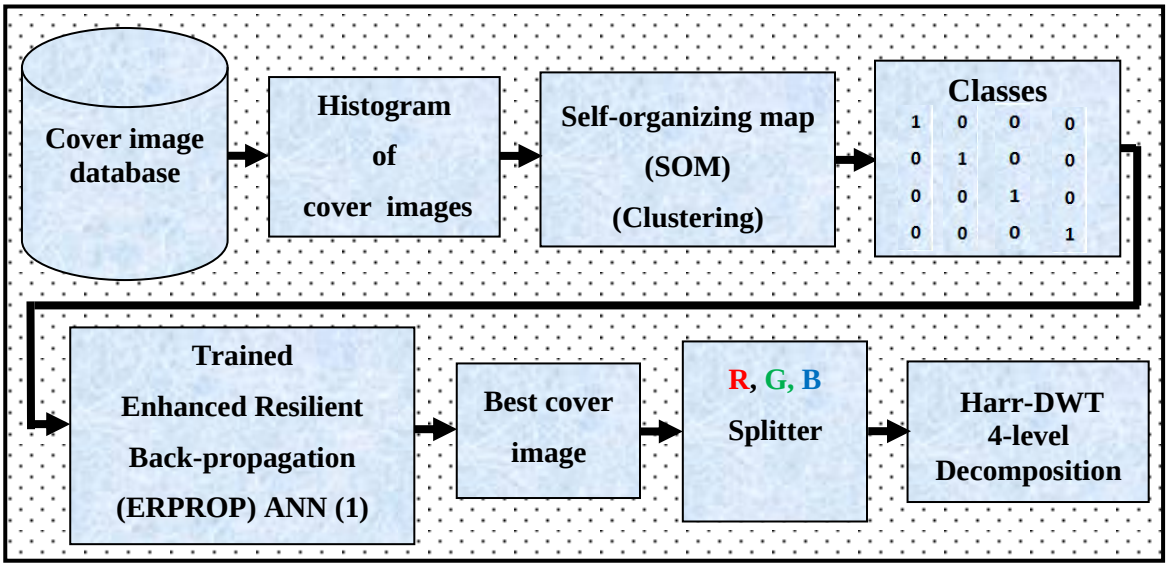


Figure 5.7: Best cover image selection and processing using hybrid of (SOM) and (ERPROP)

SOM will be trained to obtain the desired outputs of the enhanced resilient back-propagation neural network. Referring to figure 5.7, the histograms of all the cover images database is obtained first, then they are used as inputs for the SOM neural network. SOM will categorize the histograms into pre-defined classes, ([**1 0 0 0**], [**0 1 0 0**], [**0 0 1 0**], [**0 0 0 1**]), these classes will be used as desired outputs of the enhanced resilient back-propagation, whereas the histograms will be used as training patterns.

The enhanced resilient back propagation neural network will be trained and tested until it reaches to an optimal MSE with minimum number of iterations. Once the enhanced resilient back propagation neural network reaches to a stable desired behaviour, the histogram of the secret image will be used as an input to the trained enhanced resilient back-propagation neural network to get the best cover image as an output.

A. Enhanced Resilient Back-propagation Algorithm Training

(Riedmiller & Braun 1993) used the following fragment of pseudo code which represent the core of the (RPROP) learning process.

$For\ all\ weights\ and\ biases\{$

$$\boldsymbol{if}\left(\frac{\partial E}{\partial w_{ij}}(t-1)*\frac{\partial E}{\partial w_{ij}}(t)>0\right)\boldsymbol{then}\{$$

$$\Delta_{ij}(t)=\boldsymbol{minimum}\left(\Delta_{ij}(t-1)*\eta^{+},\Delta_{\max}\right)$$

$$\Delta w_{ij}(t)=-\boldsymbol{sign}\left(\frac{\partial E}{\partial w_{ij}}(t)\right)*\Delta_{ij}(t)$$

$$w_{ij}(t+1)=w_{ij}(t)+\Delta w_{ij}(t)$$

$$\}$$

$$\boldsymbol{else\ if}\left(\frac{\partial E}{\partial w_{ij}}(t-1)*\frac{\partial E}{\partial w_{ij}}(t)<0\right)\boldsymbol{then}\{$$

$$\Delta_{ij}(t)=\boldsymbol{maximum}\left(\Delta_{ij}(t-1)*\eta^{-},\Delta_{\min}\right)$$

$$w_{ij}(t+1)=w_{ij}(t)-\ \Delta w_{ij}(t-1)$$

$$\frac{\partial E}{\partial w_{ij}}(t)=0$$

$$\}$$

$$\boldsymbol{else\ if}\left(\frac{\partial E}{\partial w_{ij}}(t-1)*\frac{\partial E}{\partial w_{ij}}(t)=0\right)\boldsymbol{then}\{$$

$$\Delta w_{ij}(t)=-\boldsymbol{sign}\left(\frac{\partial E}{\partial w_{ij}}(t)\right)*\Delta_{ij}(t)$$

$$w_{ij}(t+1)=w_{ij}(t)+\ \Delta w_{ij}(t)$$

$$\}$$

$$\}$$

Where minimum and maximum operators deliver the minimum and maximum of two numbers; the sign operator deliver the sign of the argument where it returns +1 if it is positive and -1 if it has negative argument and 0 otherwise.

(Naoum, et al., 2012) proved that adding a parameter ξ to the weight updating rules $w_{ij}(t+1)=w_{ij}(t)+\ \boldsymbol{\xi}\,\Delta w_{ij}(t)$ in the pseudo code fragment above will enhance the resilient back-propagation neural network to achieve less MSE in less number of iterations. In order to find the optimal value of ξ where ($0 < \xi < 1$), we use the trial and error approach as best illustrated in the following subsection.

B. Best Learning Parameter (ξ) to Enhance (RPROP).

In addition to the Neural network parameters shown in table (5.1), we used learning parameter ξ in (ERPROP) training for both: (ERPROP) training to get best cover image, and (ERPROP) training to get the best embedding threshold.

Table (5.1): (ERPROP) neural networks parameters

Parameters	(ERPROP) of Best Cover image Selection	(ERPROP) of Best Threshold (rT, gT, bT) Selection
Input Neurons	**256**	**(CoverImg_size/2)* (CoverImg_size/2)*3*4**
Output Neurons	**4**	**1**
Number of Hidden Layer	**1**	**1**
Hidden Neurons	**156**	**100**
Transfer function	**Sigmoid**	**Sigmoid**
Number of Iterations	**100**	**100**
Slope of sigmoid	**-0.005**	**0.0005**

In the enhanced resilient back-propagation neural network (ERPROP) that is trained to choose the best cover image, we first built the network without using ξ (i.e. set ξ =1 in equation $w(t+1) = w(t) + \xi\Delta w(t)$) with initial weights set to : $-0.5 \leq$ weights ≤ 0.5 and we get MSE = 0.0733 on average. Then we set (ξ) parameter at values between 0 and 1 ($0 < \xi < 1$) in the equation ($w(t+1) = w(t) + \xi\Delta w(t)$), where we reach to the optimal value of $\xi = 0.6$ whereas in case of the enhanced resilient back propagation neural network that was used to choose the best embedding thresholds for each color layer (rT, gT ,bT), we get optimal $\xi = 0.7$ and for both cases, we get better average MSE than that got by (ERPROP) without ξ.

C. Best Cover Image (RED, Green, Blue) Splitting

After the best cover image is chosen by the trained enhanced resilient back propagation neural network, it will be split into its (R,G,B) layers, as shown in the figure 5.8.

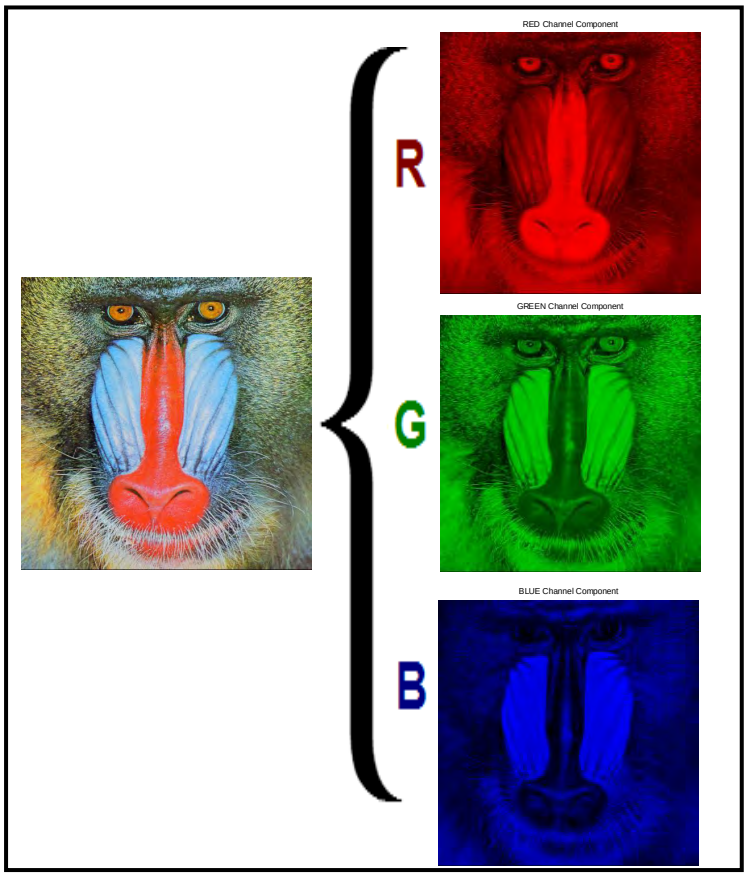


Figure 5.8: RGB layers separation of best cover image

D. Discrete Wavelet Decomposition of best Cover Image

After the cover image and secret image has been split into three color layers (R,G,B), the next step is to apply 4 level-Discrete Wavelet Transform separately to each color layer. Each layer (R,G,B) of cover-image will be decomposed into four levels and each level with various multi-resolution sub bands (Approximate, Horizontal, Vertical and Diagonal), using Haar function as mother wavelet. The aim of decomposition is to separate the low frequency components, which has the most energy of the image (Approximation), from high frequency components (Details). Figure 5.9 illustrates this

step, where it is clear that the energy is mostly concentrated in Low sub band and the other sub bands represent its reflection of details image.

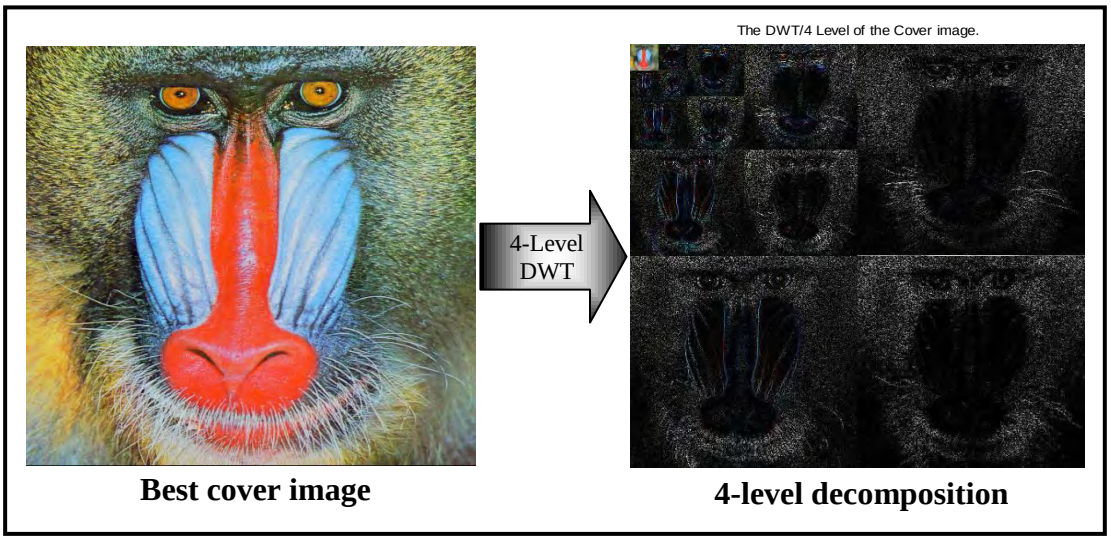


Figure 5.9: 4- Level Haar- DWT decomposition of the full color (RGB) cover

Our proposed system will embed a color layer of secret image into its corresponding layer of cover image, so the (DWT) decomposition will be applied for each layer of best cover image, and this is illustrated in figure 5.10.

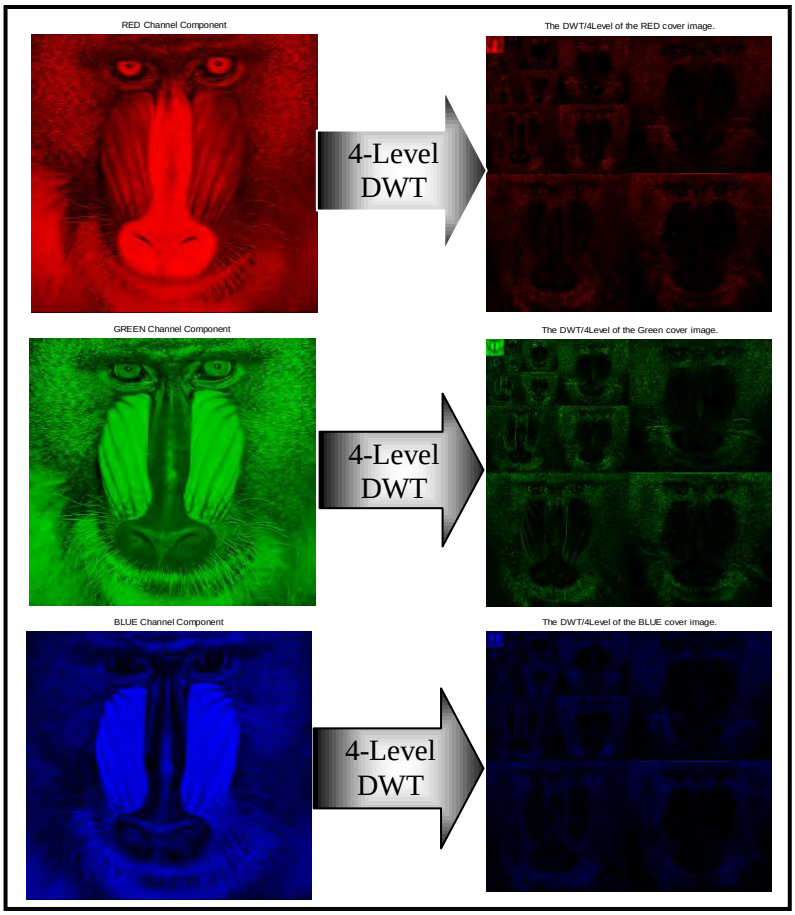


Figure 5.10: 4-Level decomposition of Red, Green and Blue layers of cover image

5.2.3 Best Embedding Threshold Selection Stage

In this stage of pre-embedding phase, the best embedding threshold parameter (T) is selected using the enhanced resilient back prorogation neural network once again. This parameter is considered the reference level that determines the availability to use the coefficients of sub bands in cover image to hide sub band bit streams coming from the secret image without losing the information in the extraction process. Figure 5.11 shows the steps involved in this stage.

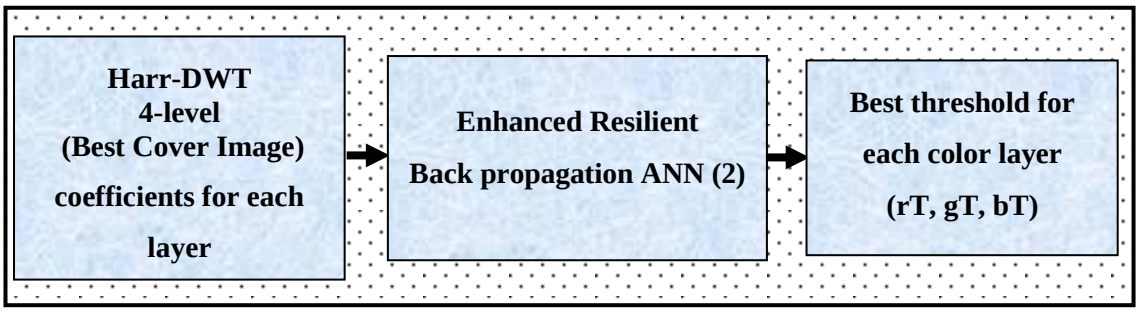


Figure 5.11: Best embedding threshold selection stage block diagram

The embedding threshold determines the size (the space) of the redundancy in the best cover image coefficients that can be used to embed the secret image. This embedding threshold can be obtained analytically by using statistical equation (5.1).

$$T = \frac{\alpha}{N} \sum_{i=0}^{N} |C_i| \qquad (5.1)$$

Where:

T: represents the embedding threshold value.

α : values range from (0-1) attenuates embedding threshold value.

C_i : the coefficients of the DWT for the cover image (Al-Ataby & Al-Naima, 2010).

However, this equation depends on the statistical characteristics of the coefficients and depends on the (α) value chosen such that it suits the case under study.

But our proposed embedding and extraction algorithms depend on using same threshold value in both embedding and extraction stages without using further locations

in the DWT coefficients to store the indices of the locations that used to store the bit streams of secret image sub bands. We proposed to use the learning power of the enhanced resilient back-propagation neural network to approximate the appropriate best threshold parameter that suits our proposed embedding algorithm.

Our resilient back-propagation neural network was trained using the normalized DWT coefficients as inputs and the best threshold value as the desired output for each layer of the cover image, the best threshold value (T) is determined after multiple trials and errors to determine the best threshold for each cover image of database that contains (100) cover images, and we reach to MSE down to ($1.4525x10^{-4}$) in (100) epochs of training. Figure 5.12 demonstrate the best embedding threshold selection stage.

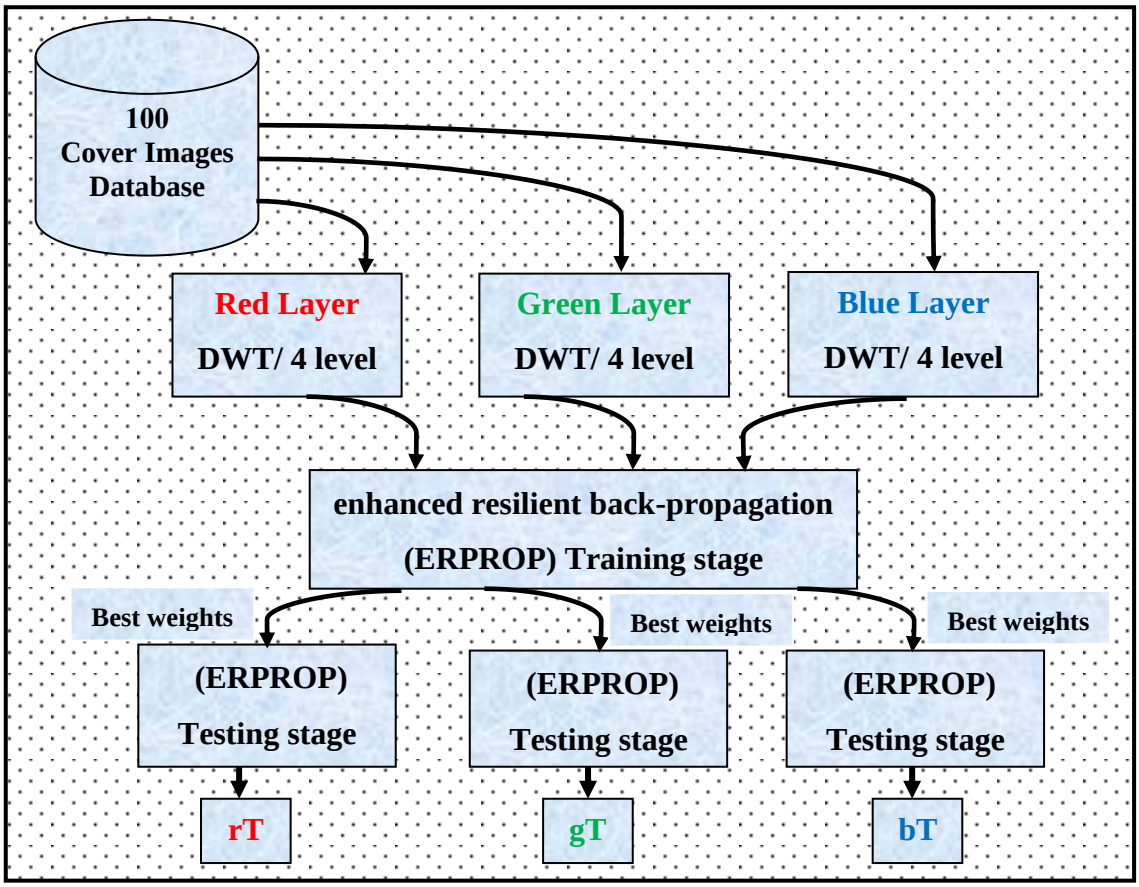


Figure 5.12 Enhanced resilient back propagation training process to select best embedding threshold block diagram

5.3 Embedding Phase

Once the results coming out of pre-embedding stages are being ready, the embedding phase can take place. In this phase, the bit stream of each sub band in the secret image will be embedded in the (DWT) coefficients of the cover image such that we embed the coefficients of one layer in the secret image in the corresponding layer in the cover image and the bit stream of one sub band in the secret image will be embedded in the corresponding sub band in the cover image and this is best illustrated in the figure 5.13.

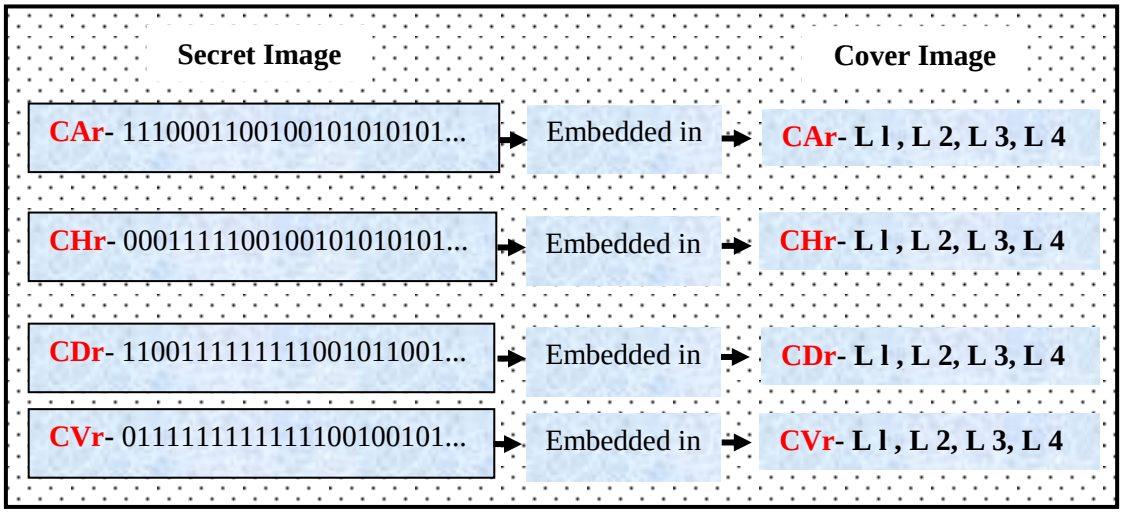


Figure 5.13: Approximation sub band of secret image embedding in the approximation sub band of cover image

Now, the coefficients of each sub band in the cover image are converted to a vector composed of the coefficients coming out of all levels and in a concatenated way. The first three values of each approximate sub band coefficients vector of each color layers are preserved for the secret key which are: seed value, embedding threshold (T), and the secret image size. The embedding then begins at the fourth coefficient of sub band vector.

Now, each coefficient is compared with the embedding threshold (T). If it is greater than threshold, then it is neglected and it will not be involved in the embedding

process. However, if the value of the coefficient is less than or equal to the embedding threshold (T), then the coefficients is converted to 16 bits binary number and then we use the least four significant bits (LSB) of this binary number to store four bits block coming out the bit stream of secret image. After the substitution, the coefficient used in embedding is transferred to its float value once again.

It's important to note that we do not use the MSB in the LSB substitution in embedding process. However, we divided the total bit stream of sub band of secret image into blocks where each block of four bits length and these blocks will be substituted in the least four significant bits of the cover image coefficients that lie below threshold parameter (T).

We have seed value and embedding threshold for each layer and we embed the coefficients of one color layer in the coefficients of the corresponding color layer and then we embed the bit stream of each sub band of the secret image in the corresponding sub band of the cover image. All of these operations have increased the layers of security of our proposed system to a very high level.

After we embed the whole bit stream of secret image sub bands in the available coefficients that lies under threshold of cover image, we apply the inverse discrete wavelet transform (IDWT). These are illustrated in Figures 5.14 (a) and (b) respectively.

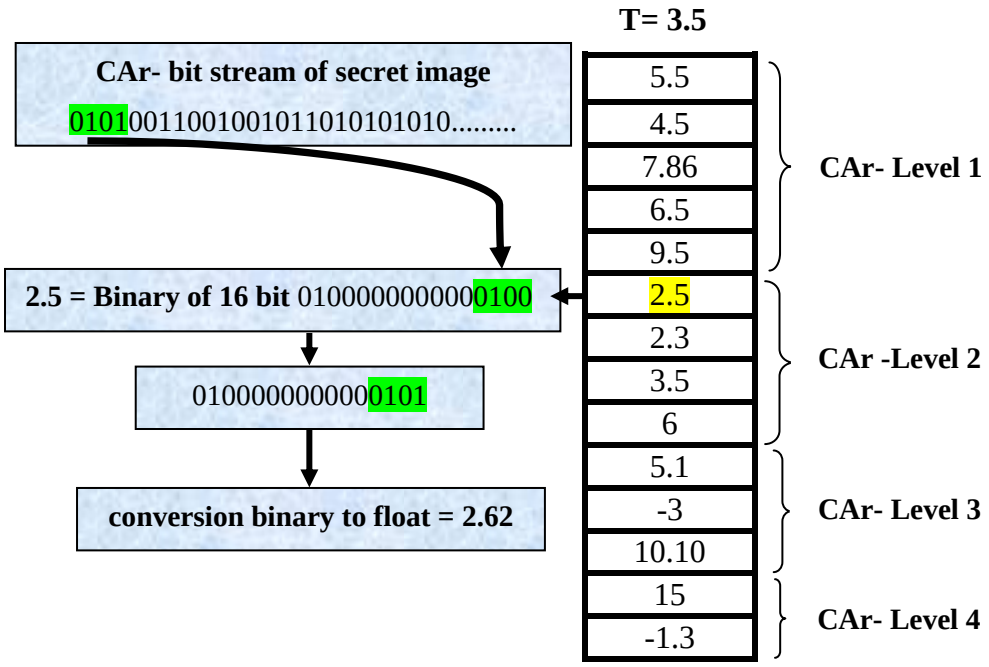


Figure 5.14 (a): Example depicting the embedding operations of CAr of secret image in CAr of cover image

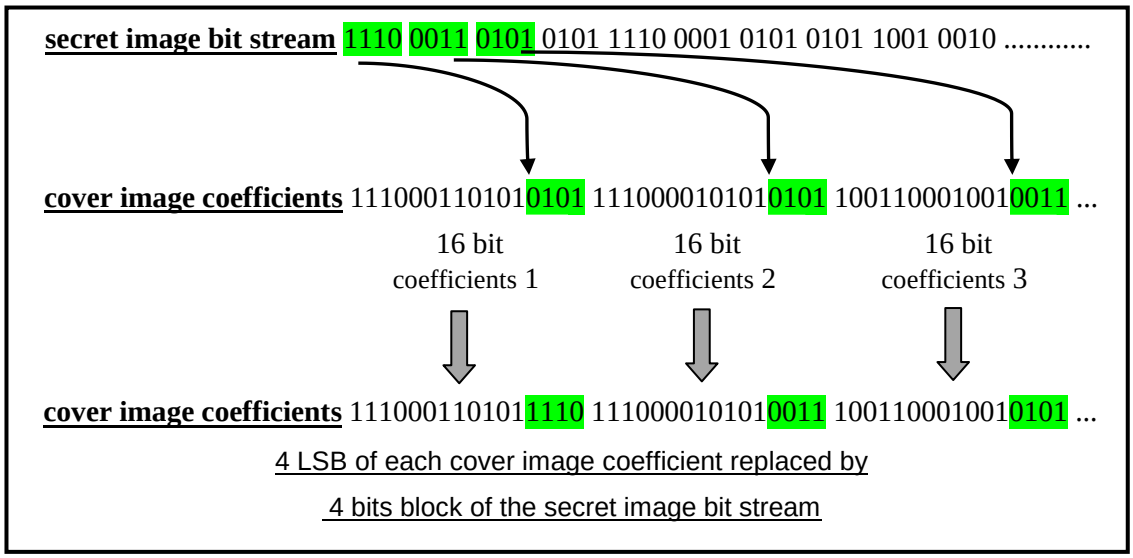


Figure 5.14 (b): Example depicting the operations steps of LSB substitution

The pre-embedding stages and the embedding phase are shown in figure 5.15.

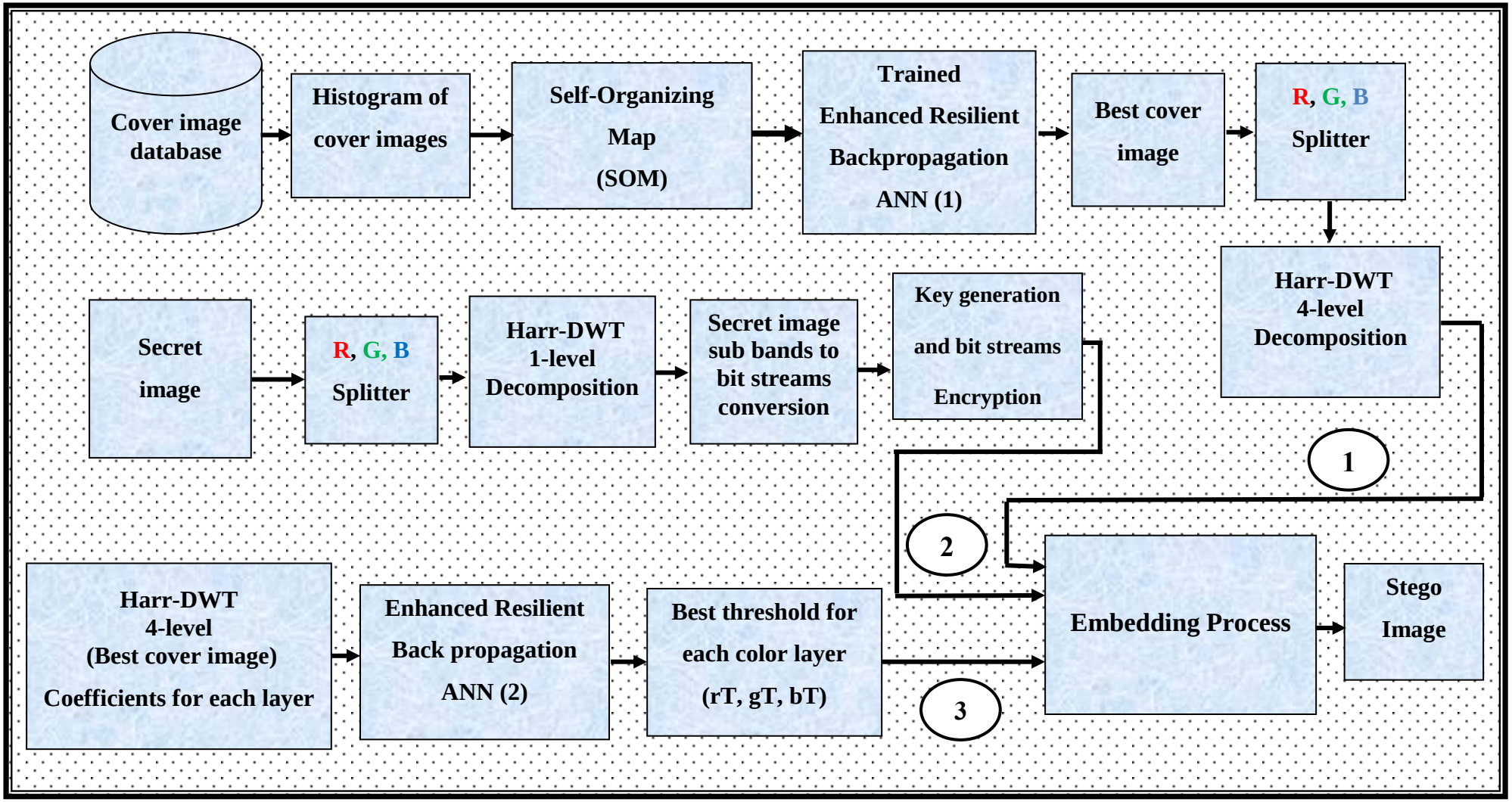


Figure 5.15: Original proposed embedding model

However, our proposed embedding model have a drawback from the elapsed time perspective, where the training process of three neural networks that were used to get the best embedding threshold for each color layer leads to massive time and computational power consumption, so we use the statistical equation (5.1) to calculate the embedding threshold instead of training three neural networks. Although we scarify the accuracy and the smartness yielded by enhanced resilient back propagation neural networks, using equation (5.1) keeps the objective tests values (PSNR and MSE) at satisfying levels, so the proposed embedding model is modified as illustrated in figure 5.16.

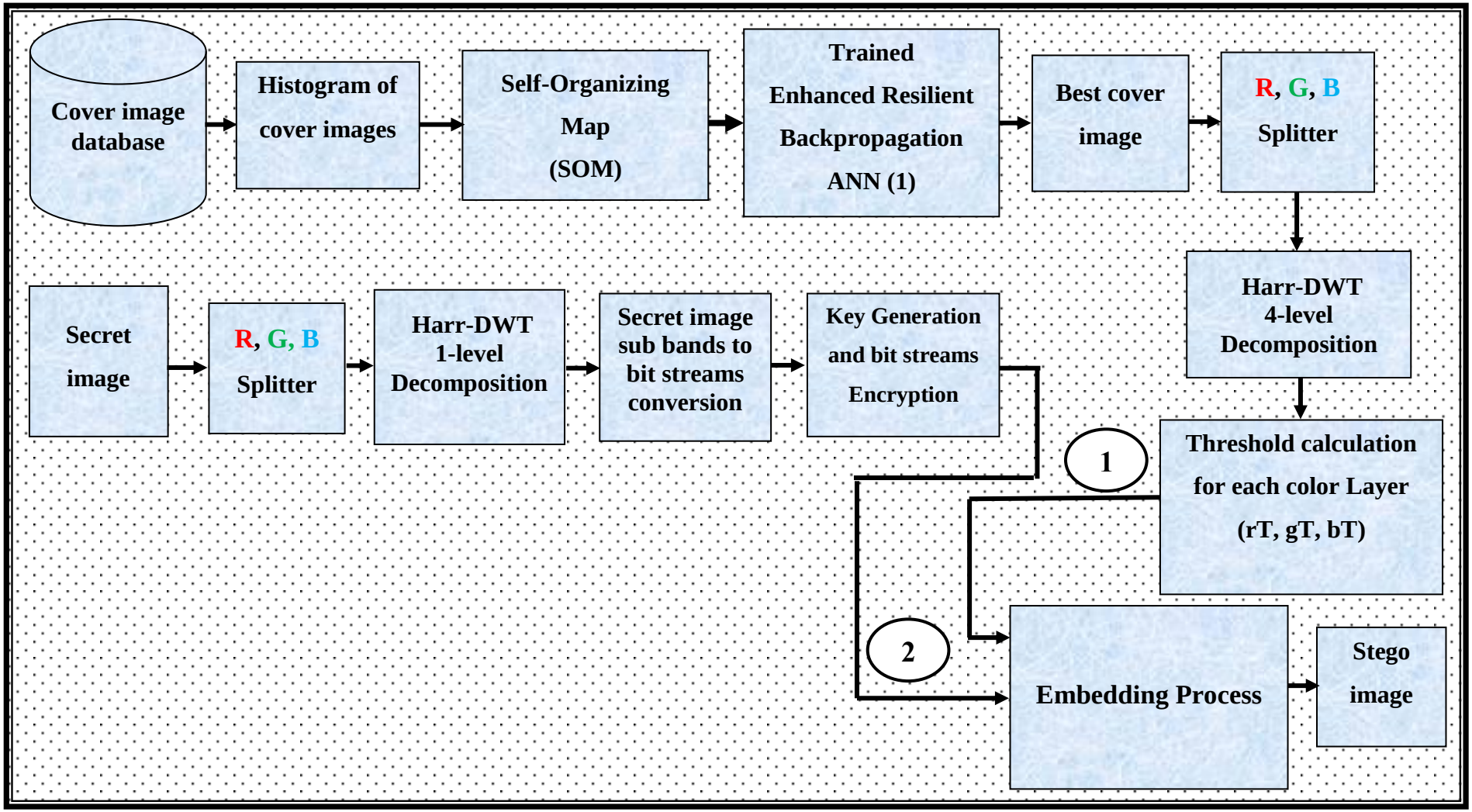


Figure 5.16: The modified proposed embedding model

5.4 Embedding Process

Inputs: Best cover image, Secret image and Embedding thresholds (rT, gT, bT).

Outputs: Stego image.

Step1: Split the cover and secret image into their (R,G,B) color layers.

Step2: Get a color layer of cover image and the same color layer of secret image.

Step3: Apply the Harr-DWT to decompose the color layer of secret image into 1- level of four non-overlapping multi-resolution sub-bands: LL1, HL1, LH1, and HH1.

Step4: Apply the Harr-DWT to decompose the color layer of cover image into 4-level of four non- overlapping multi-resolution sub-bands.

Step5: Convert the (DWT) coefficients of each sub band of secret image into 4 vectors to be ready for binary conversion process.

Step6: Convert the secret sub band vectors into 4 bit streams (bit stream for each sub band vector).

Step7: Calculate the embedding threshold of cover image color layer according to equation (5.1).

Step8: Embed the secret key: (seed value, color layer threshold (T) and the size of secret image) in the first three coefficients of the approximate sub band.

Step9: Generate the encryption key using the modified Fibonacci Linear Feedback Shift Register.

Step10: Encrypt the bit streams that obtained in step (6) using the XOR operation with the encryption key obtained in step (9).

Step11: Divide the encrypted bit streams obtained in step (10) into 4 bits-blocks.

Step12: Compare each value in the sub band vector of the cover image with the threshold value, if it is greater than the threshold (T) then, ignore it, if it is less or equals to threshold, then go to step (13).

Step13: Convert the coefficient to binary number of 16 bit length and replace the 4 Least Significant Bits with 4 bits-block coming from step (11).

Step14: Repeat steps (12) and (13) until all bits streams of the secret image are embedded in the all corresponding sub bands of cover image.

Step15: Apply the (IDWT) to get the color layer of stego-image.

Step16: Repeat the steps (2) to (15) to get the rest color layers of stego image.

Step17: Combine the color layers coming from step (16) to get the full color (RGB) stego image.

Figure 5.17 depicting the embedding algorithm flow chart of hiding red color layer of secret image in the corresponding red color layer of cover image.

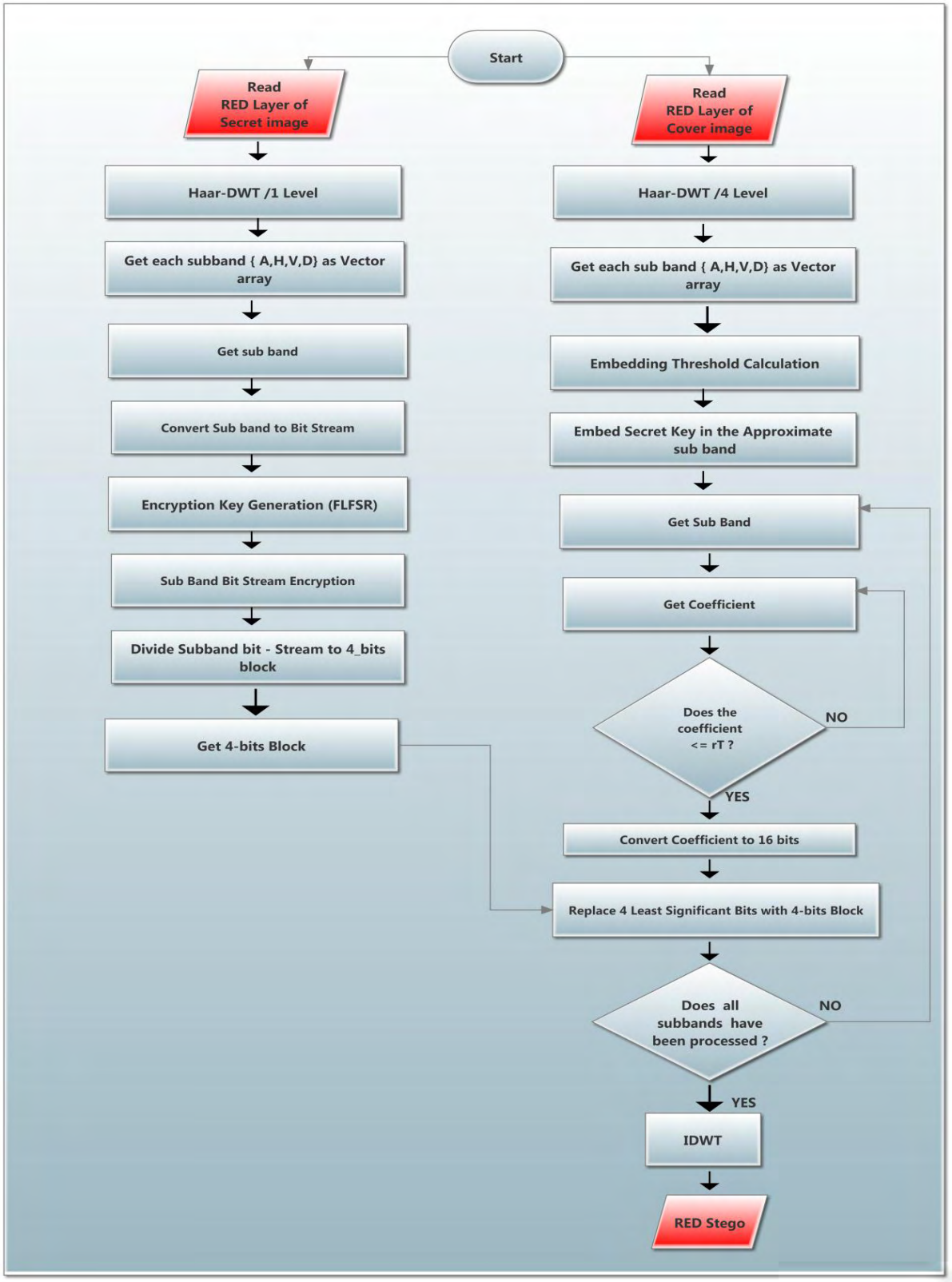


Figure 5.17: Embedding algorithm flowchart of embedding red color layer of secret image in the corresponding red color layer of cover image

5.5 Extraction Phase

Once the stego image is received by the receiver side, it is processed to extract the full color secret image, as shown in figure 5.18 below:

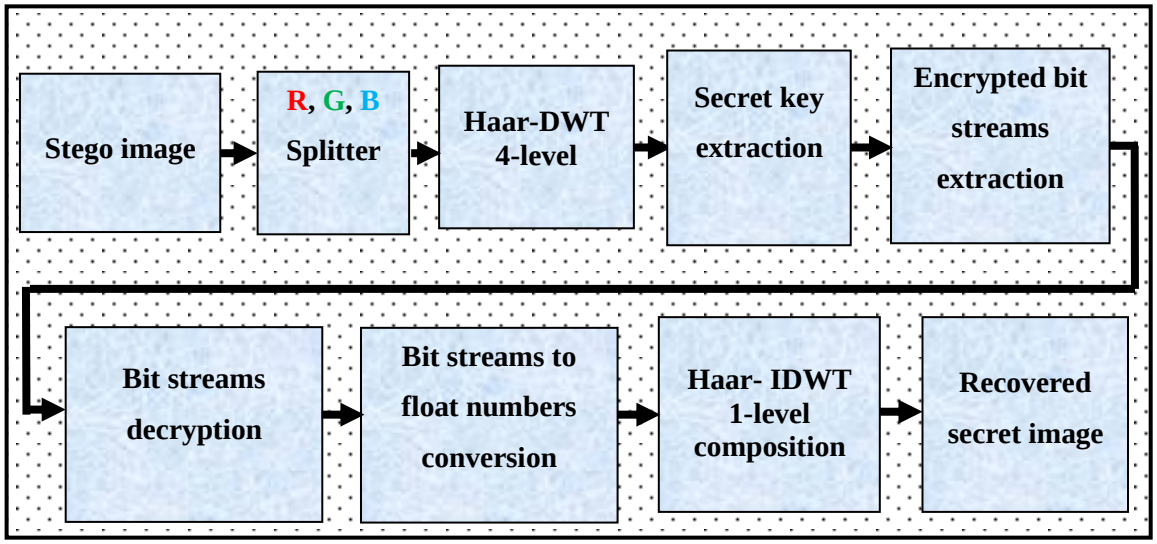


Figure 5.18: Proposed extraction model

The first step of the extraction process is to separate the stego image into its color layers (R,G,B) and then each color layer will be processed separately to get the color layers of secret image. Then these color layers will be combined together to get the full (RGB) recovered secret image and this is considered the first security layer of our proposed system.

Then each color layer of stego image is decomposed into 4 level/ DWT to get the stego image sub bands that hide the secret image bit streams. We begin with the approximate sub band, where we extract the secret key which consists of embedding threshold, seed value, and secret image size that are embedded in the first three coefficients of approximate sub band coefficients vector. Using this secret key, we will begin the extraction process. Note that each layer has its embedding seed value and its embedding threshold, so this adds another two additional security levels to our proposed system.

We compare each coefficient with the extracted embedding threshold; if the coefficient is greater than the threshold, ignore it. If it is less or equal to the threshold, convert it to binary number and extract the 4 Least Significant Bits. We repeat this process for each coefficient less or equal to the threshold until we extract the whole bit stream of secret sub band.

Now, we do the same operation for the other sub bands where we hide each sub band bit streams of secret image in its corresponding sub bands of the cover image and this adds another security level to our proposed system.

Now, we have 4 encrypted bit streams of the secret image sub bands and need to decrypt them. The decryption is done using the same steps mentioned above in the embedding phase.

Once we get the decrypted bit streams, they will be divided to 16-bits blocks and they are converted back to vectors of float numbers. We apply one level (IDWT) to get one color layer of the recovered secret image and then apply the same procedure above to get the other layers. Finally, we combine the color layers together to get the full color (RGB) secret image.

5.6 Extraction Process

Inputs: Stego image

Outputs: Recovered secret image.

Step1: Spilt the stego image into R,G,B color layers.

Step2: Get a color layer of stego image.

Step3: Apply the Harr-DWT to decompose the color layer of stego image into 4-level of four non- overlapping multi-resolution sub-bands.

Step4: Extract the secret key: (seed value, color layer threshold (T) and the size of secret image) in the first three coefficients of the approximate sub band.

Step5: Compare each value in the sub band vector of the stego image with the threshold value (T), if it is greater than the threshold (T) then, ignore it, if it is less or equals to threshold, then go to step (6).

Step6: Convert the coefficient to binary number of 16 bit length and get the 4 Least Significant Bits.

Step7: Concatenate 4-bits blocks in one bit stream where each sub band has bit stream.

Step8: Repeat steps (5), (6) and (7) until all bits streams of the secret image are extracted.

Step9: Generate the decryption key using the modified Fibonacci Linear Feedback Shift Register.

Step10: Decrypt the bit streams that obtained in step (8) using the XOR operation with the decryption key obtained in step (9).

Step11: Divide the decrypted bit streams obtained in step (10) into 16 bits-blocks.

Step12: Convert 16-bits blocks to float numbers.

Step13: Concatenate float numbers into one vector, such that a vector to each bit stream.

Step14: Apply the (IDWT)/ 1 level to get the color layer of recovered secret image.

Step15: Repeat the steps (2) to (14) to get the rest color layers of recovered secret image.

Step16: Combine the color layers coming from step (15) to get the full color (RGB) recovered secret image.

Figure 5.19 depicting the extraction algorithm flow chart of extracting red color layer of secret image from the corresponding red color layer of stego image.

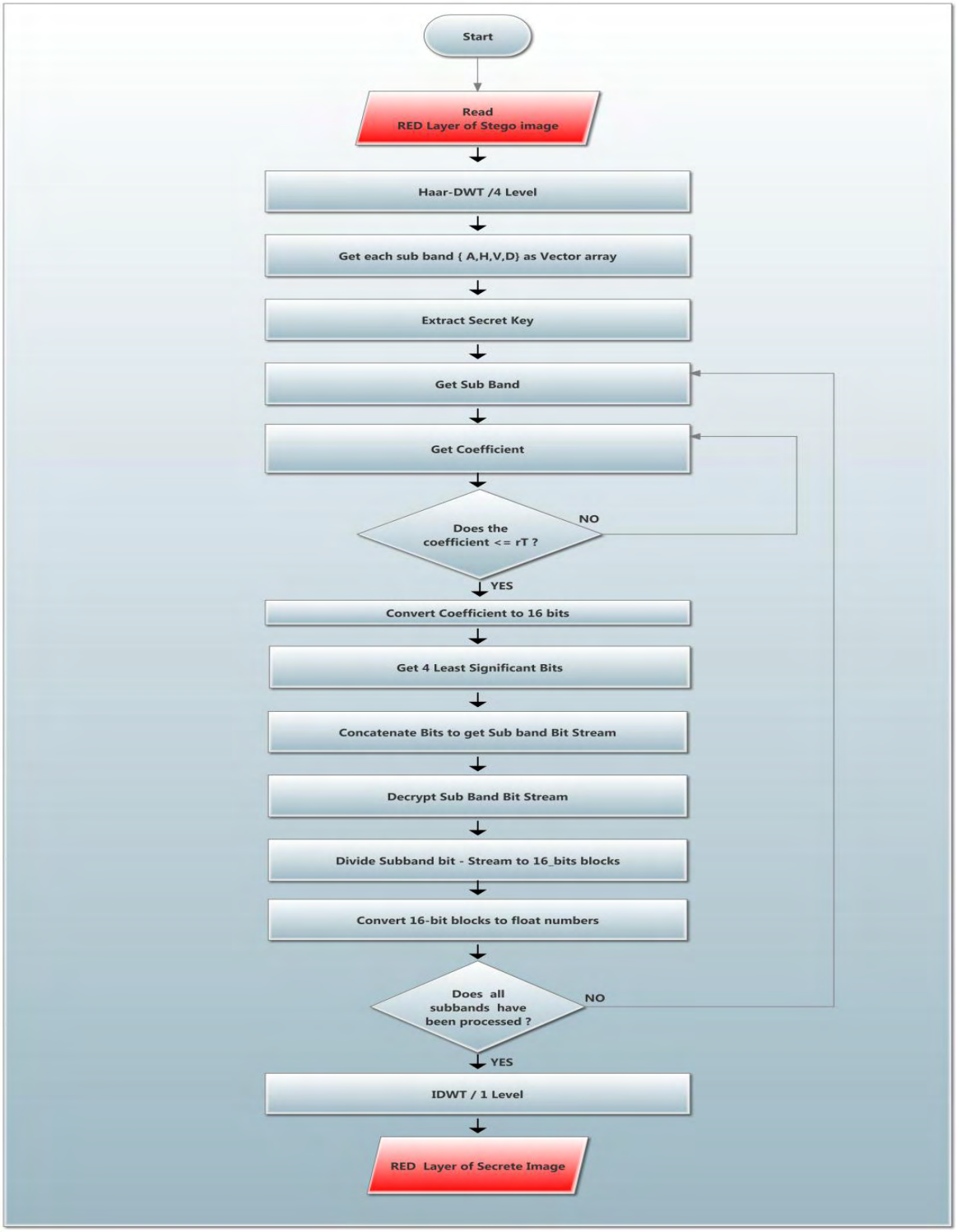


Figure 5.19: Depicting the extraction algorithm flow chart of extracting red color layer of secret image from the corresponding red color layer of stego image.

The union of the three main phases (Pre-Embedding phase, Embedding phase, and Extraction phase) that composed the proposed model is best illustrated below in figure 5.20.

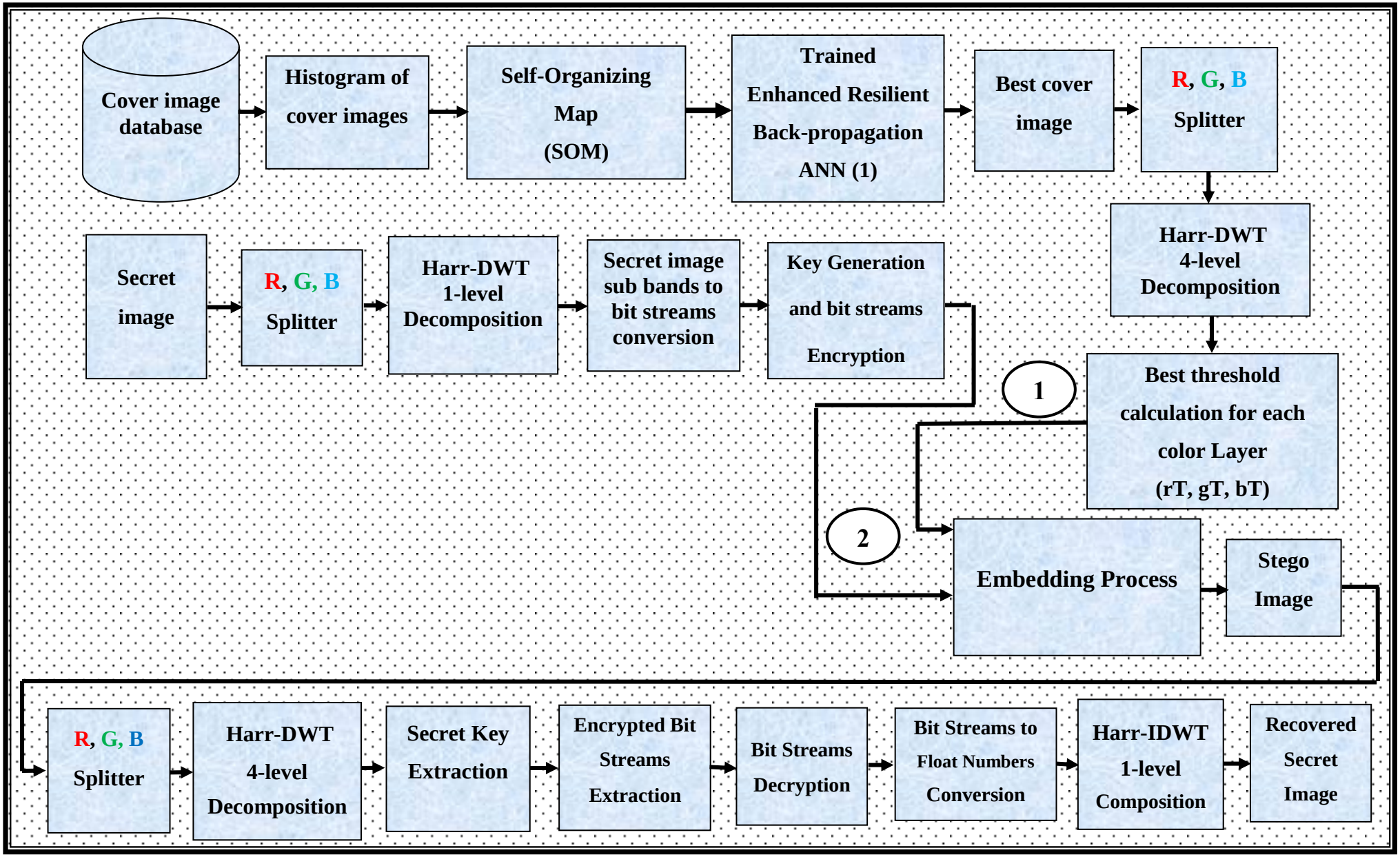


Figure 5.20: Proposed model using statistical equation (5.1)

Chapter Six

Experimental Results, Conclusion and Future Work

6.1 Implementation

This chapter presents a discussion of experimental results obtained from testing the proposed steganography system mentioned in chapter five where it was implemented using MatLab 2012a running on a Windows 7 platform. The proposed system is tested using RGB cover and secret images with different sizes. Both the secret image and the cover image are in the '.JPEG' format.

6.2 Experimental Results of the Proposed System

After running the enhanced resilient back-propagation neural network to get the best cover image, the embedding phase is then run to get the stego image and then the extraction phase is run to extract the secret image from stego image. Objective tests (PSNR and MSE) are used to evaluate the overall system performance. Figure 6.1, shows the secret images and the corresponding best cover images used to test the proposed system.

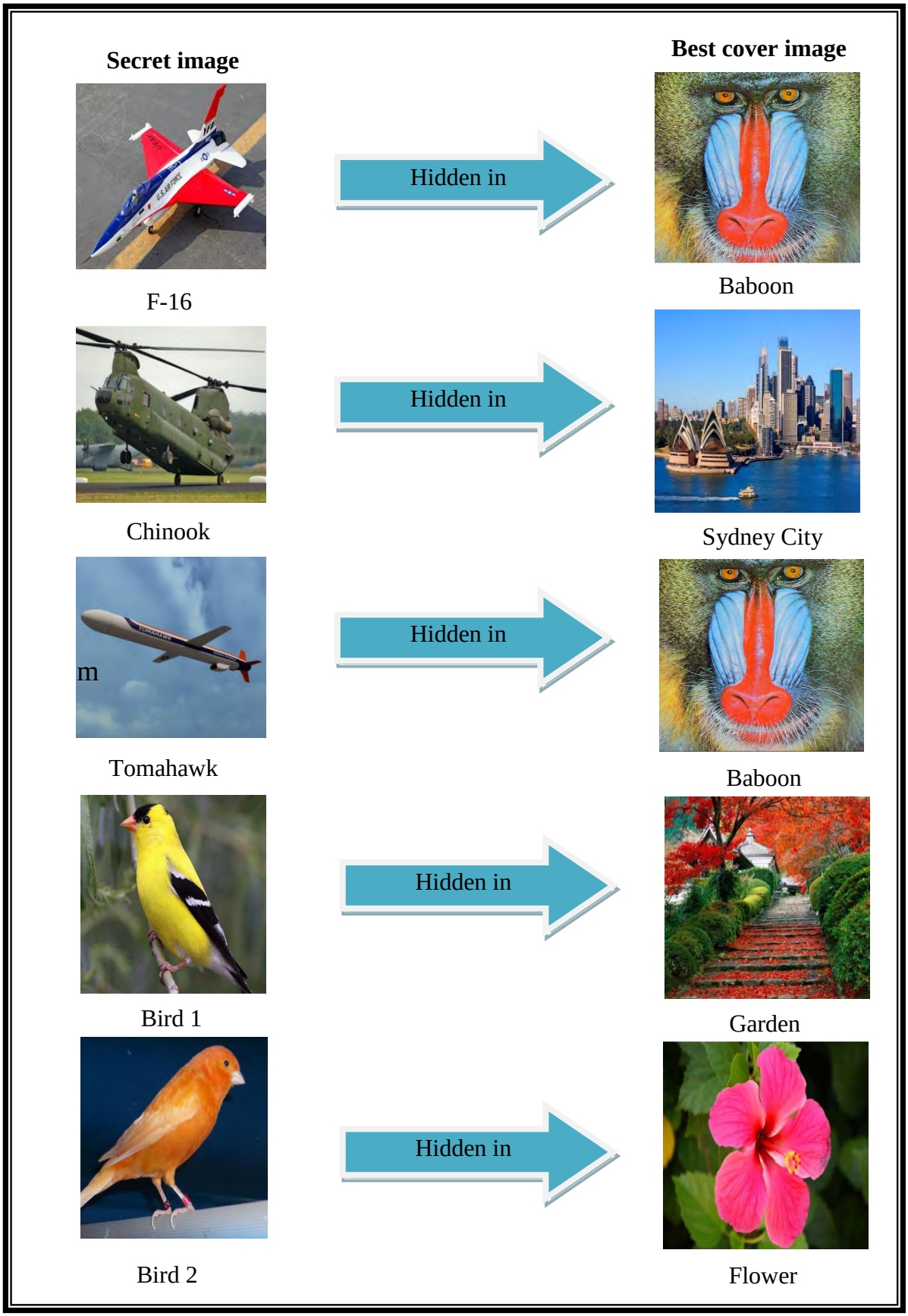


Figure 6.1: Shows the selected secret images and the corresponding best covers images chosen by (ERPROP)

6.2.1 Experimental Results of the Embedding Phase

We establish five cases, each case deals with different cover image and secret image study cases and then we tabulate the PSNR and MSE values for each case. To highlight the important characteristics of our proposed system, a histograms comparison between the resulted stego image and the cover image is presented in this section. The histogram test shows that the modified image (stego image) is not affected by the hidden image. The histogram of the cover image is approximately the same as the histogram of the resulted stego image as shown in the cases below.

Case 1: Hiding (64x64) secret image inside (256x256) cover image.

Table (6.1) lists several examples of case (1) where (64x64) secret images are hidden in (256x256) cover images.

Table (6.1): The PSNR and MSE values of case study (1)

Secret-image (64X64)	Best cover-image (256X256)	Stego-image	PSNR(dB)	MSE
F-16	Baboon	Baboon + F-16	105.8642	2.5257e-05
Chinook	Sydney City	Sydney City + Chinook	106.3960	2.3949e-05
Tomahawk	Baboon	Baboon + Tomahawk	105.9762	2.4975e-05
Bird 1	Garden	Garden + Bird	101.8378	3.7778e-05
Bird 2	Flower	Flower + Bird 2	117.8782	7.5965e-06

Figure 6.2 (a), 6.2 (b) and 6.2 (c) show hiding (Bird 2) secret image inside (Flower) cover image example.

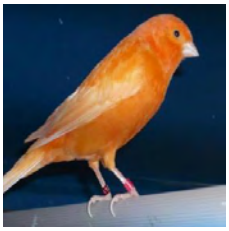


Figure 6.2 (a) : (64x64) Bird 2 secret image

Cover image

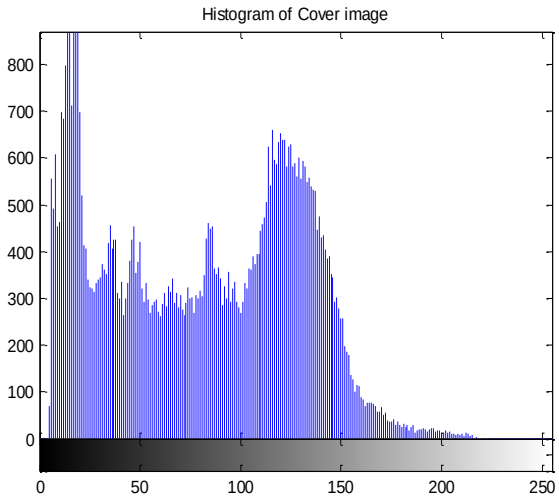


Figure 6.2 (b) : Flower cover image with its histogram

Stego image

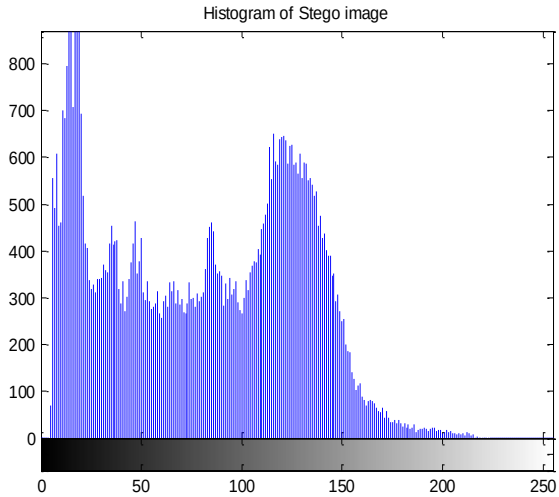


Figure 6.2 (c) : Flower stego image with its histogram

As shown in figure 6.2, our proposed system proved its high invisibility through the high perceptual transparency shown in the resulted stego image (figure 6.2 (c)) and the high similarity of histograms of both cover and stego images.

Case 2: Hiding (100x100) secret image inside (256x256) cover image.

Table (6.2) shows the examples of case (2) where (100x100) secret images are hidden in (256x256) cover images.

Table (6.2): The PSNR and MSE values of case study (2)

Secret-image (100X100)	Best cover-image (256X256)	Stego-image	PSNR/dB	MSE
F-16	Baboon	Baboon + F-16	98.4825	5.2839e-05
Chinook	Sydney City	Sydney City + Chinook	97.3963	5.8902e-05
Tomahawk	Baboon	Baboon + Tomahawk	97.8737	5.6156e-05
Bird 1	Garden	Garden + Bird 1	95.7030	6.9770e-05
Bird 2	Flower	Flower + Bird 2	111.0529	1.5033e-05

Figure 6.3 (a), 6.3 (b) and 6.3 (c) show hiding (Chinook) secret image inside (Sydney City) cover image example.

Figure 6.3 (a) : (100x100) Chinook secret image

Cover image

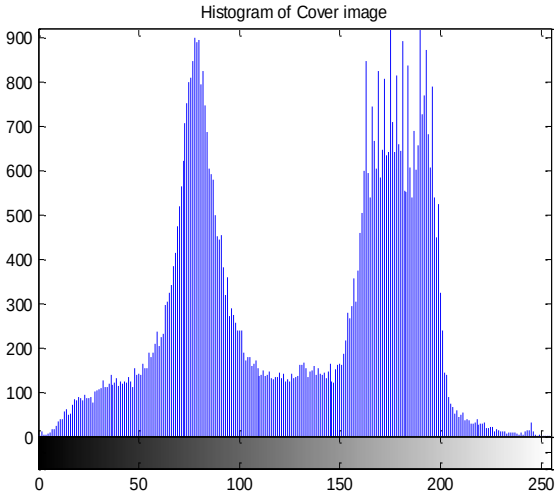


Figure 6.3 (b): Sydney City cover image with its histogram

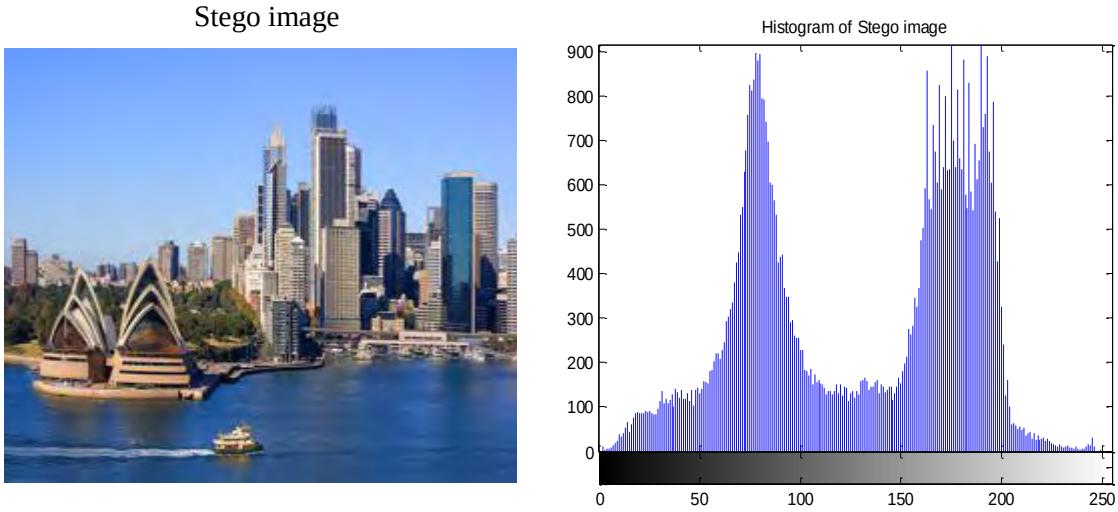


Figure 6.3 (c): Sydney City stego image with its histogram

It is apparent from figure 6.3 that although the secret image size is increased, the undetectibility of our proposed system doesn't change and the stego and cover images' histograms still keep their high similarity.

Case 3: Hiding (128x128) secret image inside (256x256) cover image

Table (6.3) shows the examples of case (3) where (128x128) secret images are hidden in (256x256) cover images.

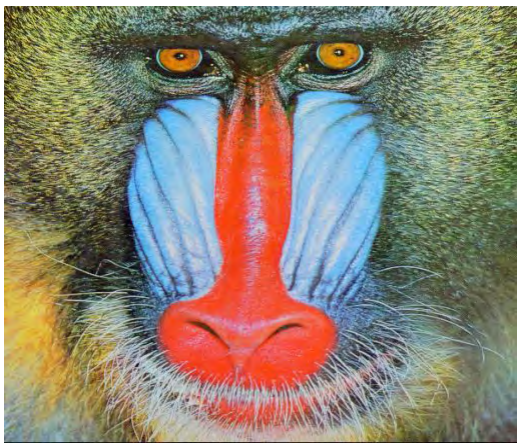
Table (6.3): The PSNR and MSE values of case study (3)

Secret-image (128X128)	Cover-image (256x256)	Stego-image	PSNR(dB)	MSE
F-16	Baboon	Baboon + F-16	98.4009	5.3273e-05
Chinook	Sydney City	Sydney City + Chinook	96.9207	6.1772e-05
Tomahawk	Baboon	Baboon + Tomahawk	98.3290	5.3657e-05
Bird 1	Garden	Garden + Bird	95.5658	7.0734e-05
Bird 2	Flower	Flower + Bird 2	112.4780	1.3036e-05

Figure 6.4 (a), 6.4 (b) and 6.4 (c) show hiding (Tomahawk) secret image inside (Baboon) cover image example.

Figure 6.4 (a): (128x128) Tomahawk secret image

Cover image



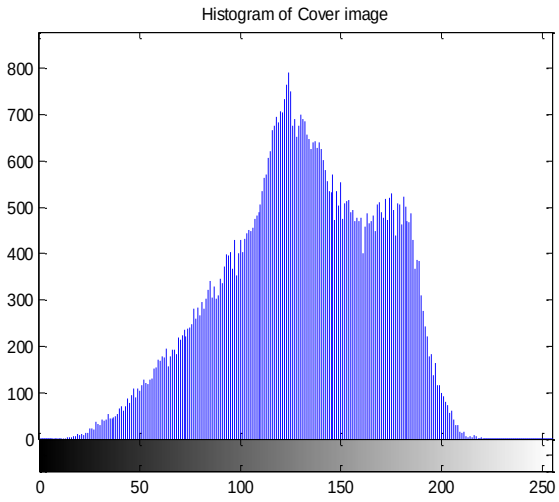


Figure 6.4 (b): Baboon cover image with its histogram

Stego image

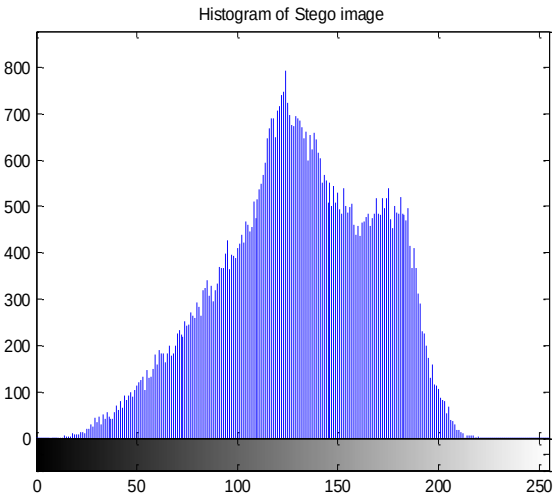


Figure 6.4 (c): Baboon stego image with its histogram

Figure 6.4 shows that our proposed system keep its high invisibility even when the secret image size becomes comparable with the cover image size without causing any visible distortion on the stego image which proved the high robustness and high capacity of our proposed system**.**

Case 4: Hiding (128x128) secret image inside (512x512) cover image.

Table (6.4) shows the examples of case (4) where (128x128) secret images are hidden in (512x512) cover images.

Table (6.4): The PSNR and MSE values of case study (4)

Secret-image (128X128)	Cover-image (512X512)	Stego-image	PSNR/dB	MSE
F-16	Baboon	baboon + F-16	123.2455	4.4414e-06
Chinook	Sydney City	Sydney City + Chinook	124.6816	3.8472e-06
Tomahawk	Baboon	baboon + Tomahawk	122.0270	5.0169e-06
Bird 1	Garden	Garden + Bird	118.7521	6.9609e-06
Bird 2	Flower	Flower + Bird 2	137.0849	1.1130e-06

Figure 6.5 (a), 6.5 (b) and 6.5 (c) show hiding (Bird 1) secret image inside (Garden) cover image example.

Figure 6.5 (a): (128x128) Bird 1 secret image

Cover image

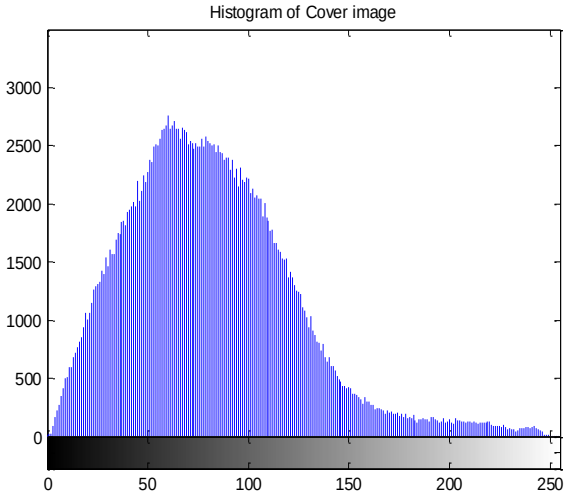


Figure 6.5 (b): Garden cover image with its histogram

Stego image

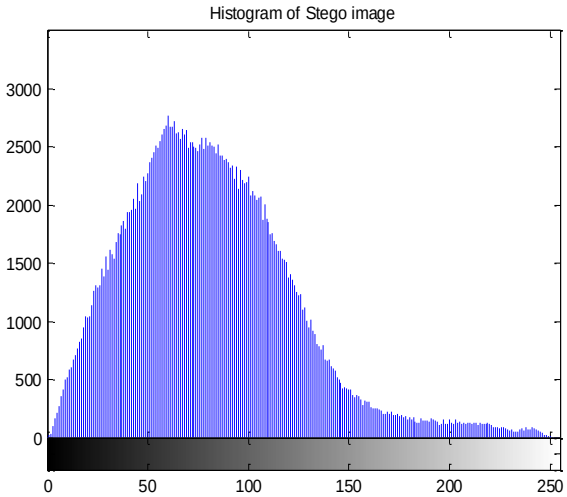


Figure 6.5 (c): Garden stego image with its histogram

Case 5: Hiding (150x150) secret image inside (512x512) cover image.

Table (6.5) lists several examples of case (5) where (150x150) secret images are hidden in (512x512) cover images.

Table (6.5): The PSNR and MSE values of case study (5)

Secret-image (150X150)	Cover-image (512X512)	Stego-image	PSNR/dB	MSE
F-16	Baboon	Baboon + F-16	120.5261	5.8293e-06
Chinook	Sydney City	Sydney City + Chinook	121.8677	5.0974e-06
Tomahawk	Baboon	Baboon + Tomahawk	119.4119	6.5164e-06
Bird 1	Garden	Garden + Bird	115.1675	9.9618e-06
Bird 2	Flower	Flower + Bird 2	133.3082	1.6237e-06

Figure 6.6 (a), 6.6 (b) and 6.6 (c) show hiding (F-16) secret image inside (Baboon) cover image example.

Figure 6.6 (a): (150x150) F-16 secret image

Cover image

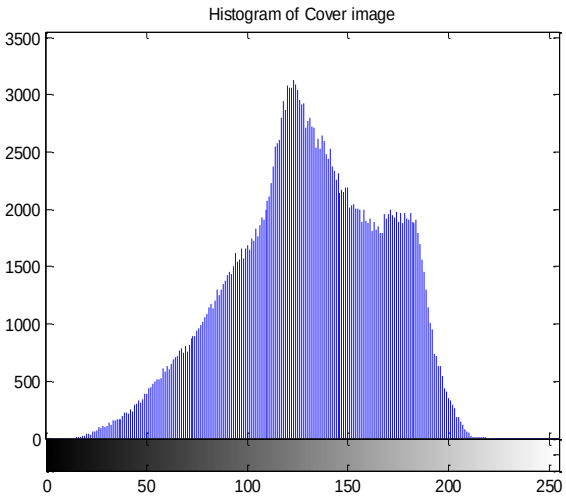


Figure 6.6 (b): Baboon cover image with its histogram

Stego image

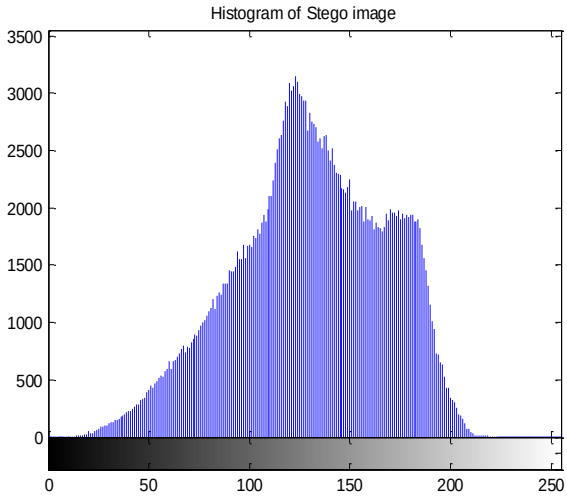


Figure 6.6 (c): Baboon stego image with its histogram

From figures 6.5 and 6.6, we conclude that histograms of stego images have strong similarity to histogram of cover images. Therefore, the attacker will not observe the difference between statistics of stego image and cover image and this will result in that the stego image has a high chance of survival.

Moreover, the sizes of both cover and stego images have the same image sizes (in bytes) or may have a slight size difference, which emphasizes the high quality of our proposed embedding model. This is best illustrated in table (6.6) that shows the image size comparison between stego and cover images in different cases.

Table (6.6): Image size (in bytes) comparison between cover and stego images

Cover-image Size (Pixels)	Stego-image Size (Pixels)	Cover-image Size (Kbytes)	Stego-image Size(Kbytes)
Baboon (256x256)	Baboon (256x256)	19.144 KB	19.170 KB
Sydney City (512x512)	Sydney City (512x512)	32.507 KB	32.511 KB
Flower (512x512)	Flower (512x512)	25.430 KB	25.449 KB

From tables (6.1) to (6.5), we conclude that the experimental results obtained by the two objective tests (PSNR and MSE) are robust, stable and imperceptible.

6.2.2 Experimental Result of the Extraction Phase

The experimental results of extracting the secret image from the stego image for the cases above are shown below. Also the histogram test is applied to the original secret image and recovered secret image in each case.

Case 1: Extract (64x64) secret image from (256x256) stego image

Table (6.7) lists several examples of case (1) where (64x64) secret images are extracted from (256x256) stego images.

Table (6.7): The PSNR and MSE values for secret image extraction case study (1)

Stego-image (256x256)	**Original Secret-image** (64X64)	**Recovered secret image** (64X64)	**PSNR (dB)**	**MSE**
Baboon + F-16	F-16	F-16	88.3651	1.4533e-04
Sydney City + Chinook	Chinook	Chinook	86.6446	1.6724e-04
Baboon + Tomahawk	Tomahawk	Tomahawk	80.6842	2.1858e-04
Garden + Bird 1	Bird 1	Bird 1	92.5407	9.5721e-05
Flower + Bird 2	Bird 2	Bird 2	92.6455	8.8155e-05

Figure 6.7 (a) and 6.7 (b) show extracting (Bird 1) secret image from (Flower) stego image example.

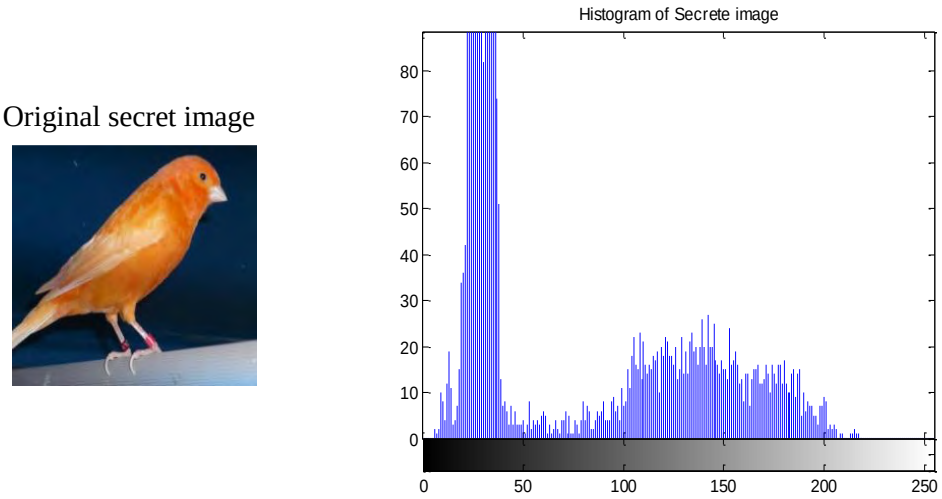


Figure 6.7 (a): Original secret image with its histogram

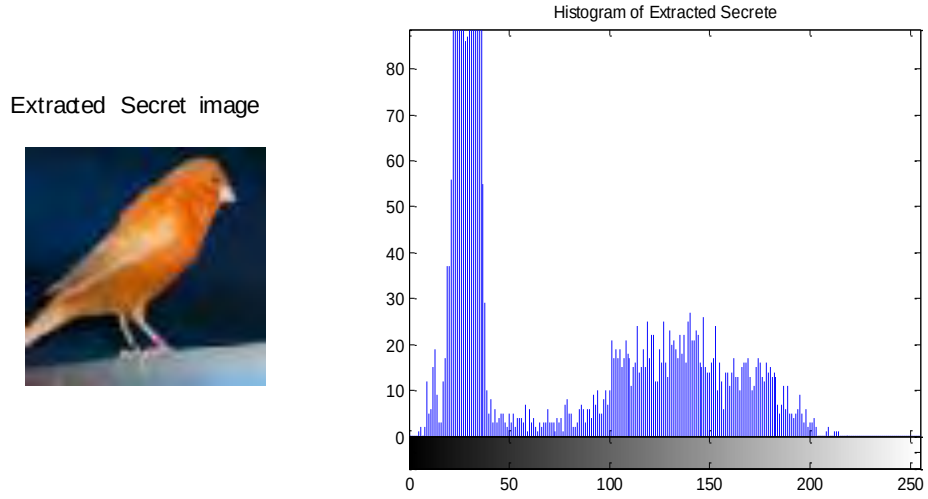


Figure 6.7 (b): Extracted secret image with its histogram

The experimental results of figure 6.7 show that the recovered secret image has almost the same histogram of the original secret image which is considered a natural result of the high visibility of the proposed system that is exemplified in the high attained PSNR.

Case 2: Extract (100x100) secret image from (256x256) stego image.

Table (6.8) shows the examples of case (2) where (100x100) secret images are extracted from (256x256) stego images.

Table (6.8): The PSNR and MSE values for secret image extraction case study (2)

Stego-image (256x256)	Original secret-image (100X100)	Recovered secret image (100X100)	PSNR (dB)	MSE
Baboon + F-16	F-16	F-16	88.1890	1.4791e-04
Sydney City + Chinook	Chinook	Chinook	87.2054	1.6320e-04
Baboon + Tomahawk	Tomahawk	Tomahawk	80.9361	2.1515e-04
Garden + Bird	Bird 1	Bird 1	92.6070	9.5089e-05
Flower + Bird 2	Bird 2	Bird 2	92.8931	8.8110e-05

Figure 6.8 (a) and 6.8 (b) show extracting (Chinook) secret image from (Sydney City) stego image example.

Original secret image

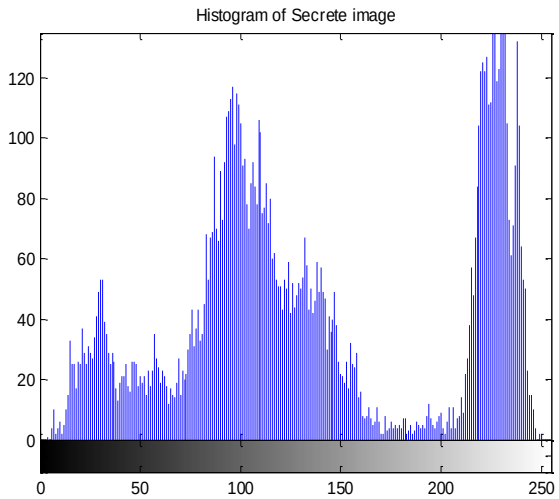


Figure 6.8 (a): Original secret image with its histogram

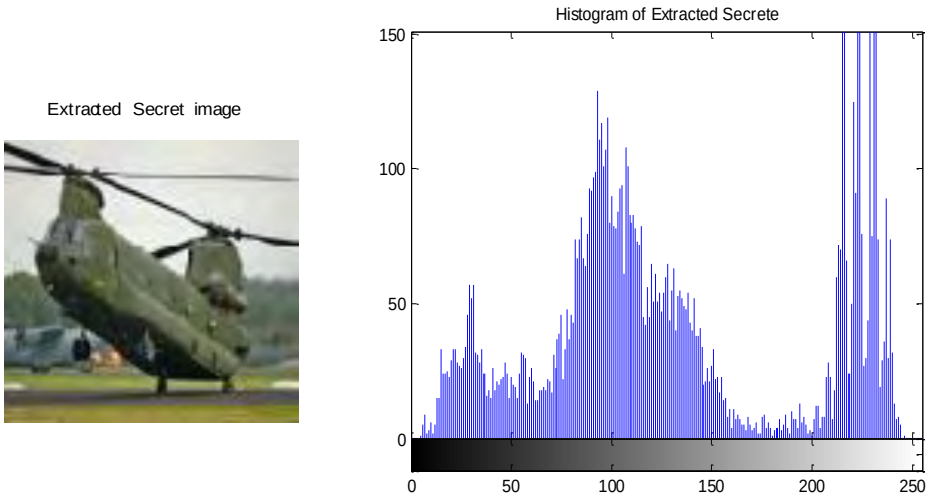


Figure 6.8 (b): Extracted secret image with its histogram

Case 3: Extract (128x128) secret image from (256x256) stego image.

Table (6.9) shows the examples of case (2) where (100x100) secret images are extracted from (256x256) stego images.

Table (6.9): The PSNR and MSE values for secret image extraction case study (3)

Stego-image (256x256)	**Original Secret-image** (128X128)	**Recovered secret image** (128X128)	**PSNR (dB)**	**MSE**
Baboon + F-16	F-16	F-16	88.3033	1.4623e-04
Sydney City + Chinook	Chinook	Chinook	87.1426	1.6423e-04
Baboon + Tomahawk	Tomahawk	Tomahawk	81.3511	2.1419e-04
Garden + Bird 1	Bird 1	Bird 1	93.1047	9.0472e-05
Flower + Bird 2	Bird 2	Bird 2	92.6791	9.0016e-05

Figure 6.9 (a) and 6.9 (b) show extracting (Tomahawk) secret image from (Baboon) stego image example.

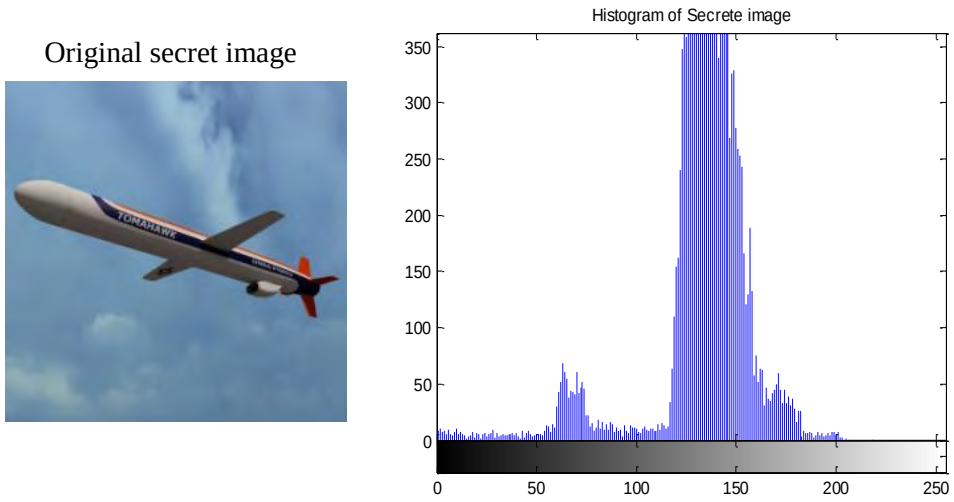


Figure 6.9 (a) Original secret image with its histogram

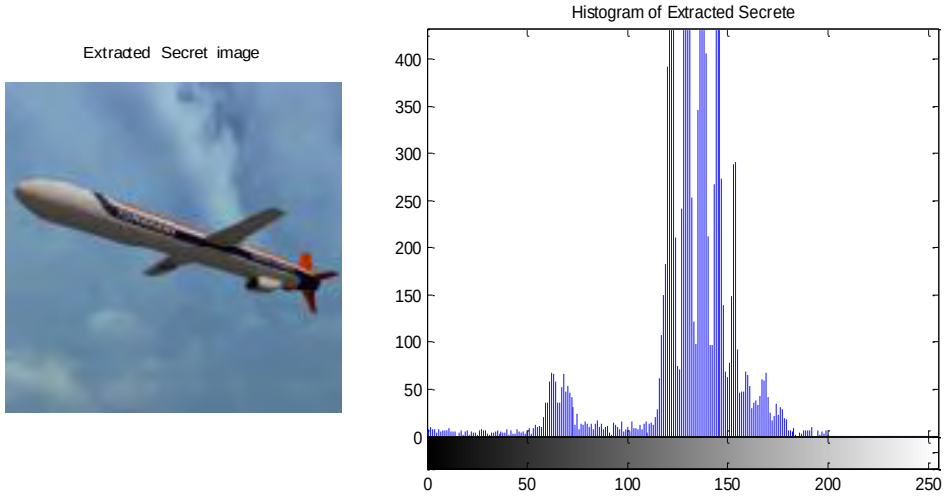


Figure 6.9 (b) Extracted secret image with its histogram

The experimental results of figure (6.9) shows that although the secret image was comparable in size to the cover image, it recovered with high PSNR and high perceptual transparency illustrated in the similar histograms of both the original secret image and the recovered one.

Case 4: Extract (128x128) secret image from (512x512) stego image

Table (6.10) shows the examples of case (4) where (128x128) secret images are extracted from (512x512) stego images.

Table (6.10): The PSNR and MSE values for secret image extraction case study (4)

Stego-image (512x512)	**Original secret-image** (128X128)	**Recovered secret image** (128X128)	**PSNR/dB**	**MSE**
Baboon + F-16	F-16	F-16	88.3033	1.4623e-04
Sydney City + Chinook	Chinook	Chinook	87.1426	1.6423e-04
Baboon + Tomahawk	Tomahawk	Tomahawk	81.3511	2.1419e-04
Garden + Bird 1	Bird 1	Bird 1	93.1047	9.0472e-05
Flower + Bird 2	Bird 2	Bird 2	92.6791	9.0016e-05

Figure 6.10 (a) and 6.10 (b) show extracting (Bird 1) secret image from (Garden) stego image example.

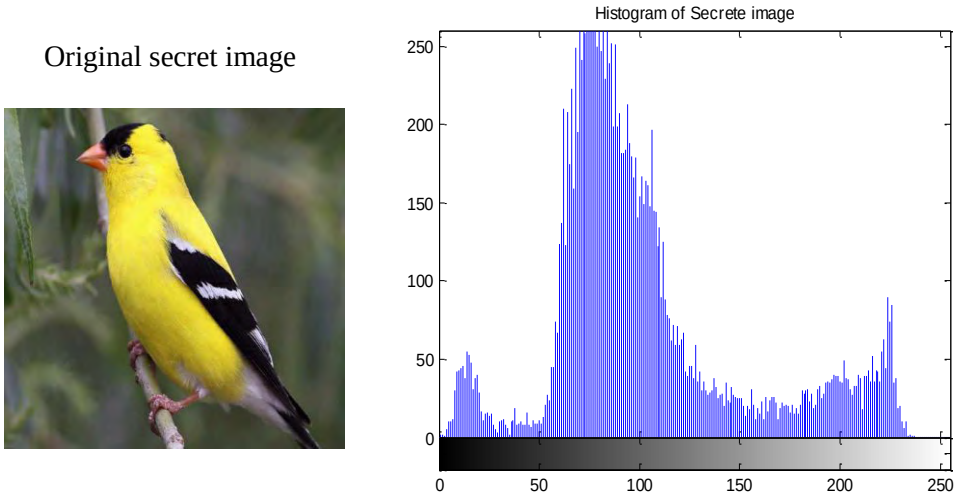


Figure 6.10 (a) Original secret image with its histogram

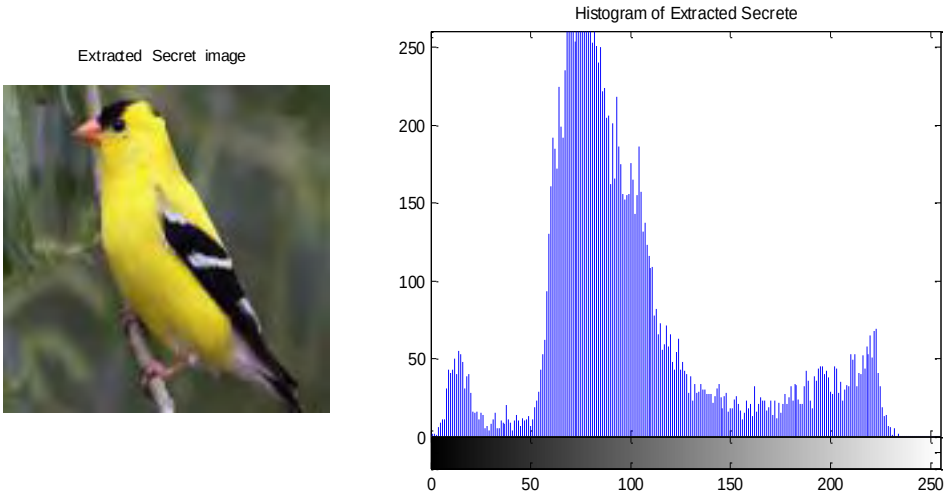


Figure 6.10 (b) Extracted secret image with its histogram

Case 5: Extract (150x150) secret image from (512x512) stego image.

Table (6.11) lists several examples of case (5) where (150x150) secret images are extracted from (512x512) stego images.

Table (6.11): The PSNR and MSE values for secret image extraction case study (5)

Stego-image (512x512)	**Original Secret-image** (150X150)	**Recovered secret image** (150X150)	**PSNR/dB**	**MSE**
Baboon + F-16	F-16	F-16	88.2296	1.4731e-04
Sydney City + Chinook	Chinook	Chinook	87.0785	1.6528e-04
Baboon + Tomahawk	Tomahawk	Tomahawk	81.3334	2.1261e-04
Garden + Bird 1	Bird 1	Bird 1	92.9127	9.1504e-05
Flower + Bird 2	Bird 2	Bird 2	93.3504	8.6211e-05

Figure 6.11 (a) and 6.11 (b) show extracting (F-16) secret image from (Babbon) stego image example.

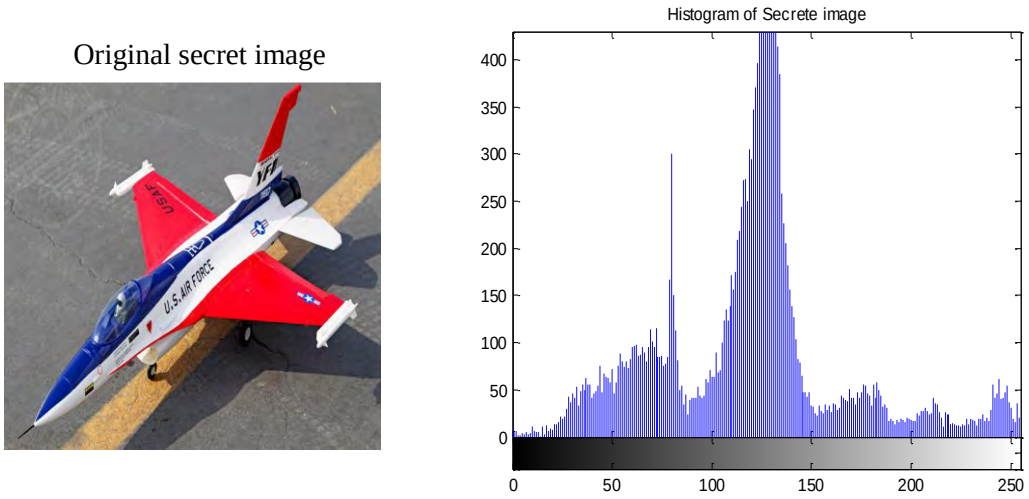


Figure 6.11 (a): Original secret image with its histogram

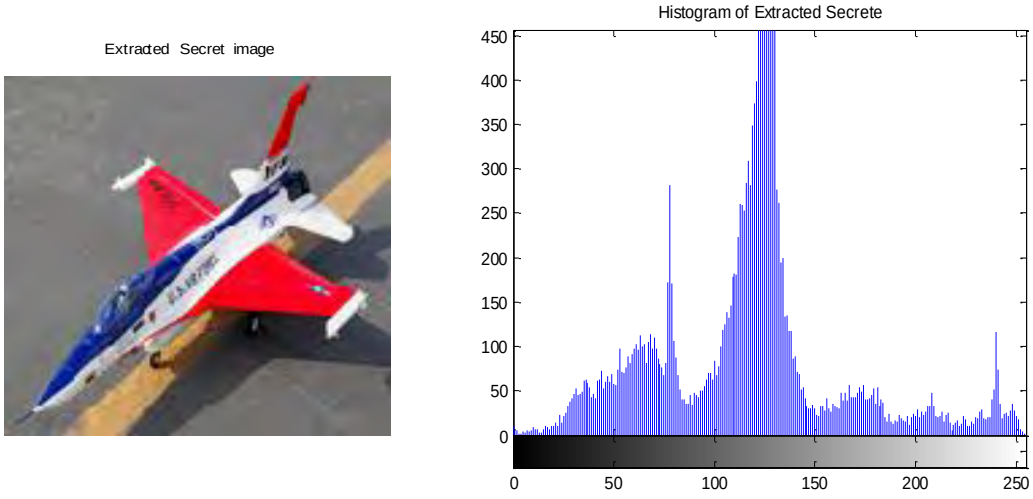


Figure 6.11 (b): Extracted secret image with its histogram

As shown in both figures 6.10 and 6.11, the proposed system was capable to recover the secret image efficiently and smoothly proved by the high attained extraction PSNR and low MSE value.

6.3 Processing Time Comparison between Original Proposed Embedding Model and Modified Proposed Embedding Model

Table (6.12) depicts the comparison of PSNR and processing time between original proposed embedding model and the modified proposed embedding model for different study cases.

Table (6.12): PSNR and Processing time of proposed embedding models (original and modified)

Case No.	Secret image	Cover image	PSNR of Original proposed model	PSNR of Modified proposed model	Processing time of Original proposed model (minutes)	Processing time of Modified Proposed model (minutes)
Case 1	Bird 2 (64x64)	Flower (256x256)	117.8782	117.8782	12.935	2.562
Case 2	Chinook (100x100)	Sydney City (256x256)	97.3963	97.3963	14.760	4.921
Case 3	Tomahawk (128x128)	Baboon (256x256)	98.3290	98.3290	18.356	7.973
Case 4	Bird 1 (128x128)	Garden (512x512)	118.7521	118.7521	66.960	8.025
Case 5	F-16 (150x150)	Baboon (512x512)	120.5261	120.5261	62.636	11.040

From table (6.12), we conclude that using statistical equation (5.1) to compute the embedding threshold (T) leads to decreasing in time.

6.4 Comparing our proposed algorithm with other algorithms

To evaluate the performance of our proposed steganography system, several simulations have been performed in order to compare its performance with other existing steganograpnhy systems. Tables (6.13) to (6.15) illustrate the comparison of PSNR of our proposed method with other methods.

Table (6.13): PSNR of our proposed method and DWT method

Method	Secret image	Cover image	PSNR (dB)
Discrete wavelet transform (Parul., Manju., & Rohil, 2014).	RGB image (256x256)	RGB image (512x512)	47.8901
Our Proposed Method	RGB image (256x256)	RGB image (512x512)	115.6178

Table (6.14): PSNR of our proposed method and DWT method

Method	Secret image	Cover image	PSNR (dB)
Discrete wavelet transform (DWT) (Naoum, et al., 2014).	RGB image (64x64)	RGB image (256x256)	39.0606
Our Proposed Method	RGB image (64x64)	RGB image (256x256)	101.8378

Table (6.15): PSNR of our proposed method and DCT method

Method	Secret image	Cover image	PSNR (dB)
Discrete cosine transform (DCT) (Naoum, et al., 2013).	RGB image (64x64)	RGB image (256x256)	29.859
Our Proposed Method	RGB image (64x64)	RGB image (256x256)	105.8642

From table (6.13) to (6.15) we conclude that our proposed method has better results comparing with other steganographic methods applying to different sizes of cover and secret images.

6.5 Conclusions

1. The selection of the best cover image using the hybrid artificial neural network (SOM and ERPROP) played an important role in improving the overall system performance.
2. Splitting the secret and cover image into (R,G,B) color layers and different level DWT decomposition led to high perceptual quality in both embedding and extraction phases.
3. The proposed system embedded the secret image in the cover image based on Haar-DWT, which provided good extracted secret image quality that led to increasing in the imperceptibility of the system.
4. The approximate and details sub bands in the proposed steganographic technique are used to hide the bit streams of secret image depending on embedding threshold (T) which resulted in high embedding capacity. The capacity is about (1/4) of the cover image in worse cases.
5. Transforming the sub bands of secret image into bits streams and hiding it in the bits of LSB position of the DWT coefficients of cover image resulted in high PSNR and smaller MSE in both embedding and extraction phases.
6. The statistical equation (5.1) dramatically decreased the time needed to compute the embedding threshold (T) in comparison with time needed by (ERPROP) to perform the same task.
7. The approximated values of embedding thresholds (rT, gT, bT) that produced using (ERPROP) is extremely close to the exact embedding thresholds that calculated using the statistical equation (5.1) due to the low MSE that reach down to ($1.2*10^{-6}$) which indicates the high accuracy of (ERPROP) operation during testing phase.
8. The strong histogram similarity of stego and cover image proves the high robustness and security of our system against attackers.

9. The proposed combination between steganography and cryptography improved the security layers of our research to competitive levels compared with existing modern steganographic systems. So it is difficult to know the original hidden image since it is encrypted before being embedded.
10. Despite of the computational complexity of this system, it is suitable for real time applications because the run time is acceptable (elapsed time =11.040 Minute) in worst cases.

6.6 Future Work

In aim of concluding our thesis, the following ideas can be recommended by the researcher:

1. Investigating the hiding of other media files such as text, audio and video clips files inside RGB images.
2. Applying 24 bits/coefficient instead of 16 bits/coefficient during the binary conversion of secret image to achieve higher PSNR and transparency.
3. Compressing the secret image before the hiding process in such a way that minimizes the amount of sent information and therefore minimizing the chance of image degradation.
4. Investigating the application of 16-bit color layer (48 bits for each pixel) to enhance the proposed technique to hide larger amount of data.
5. Further theoretical analysis is needed to find further optimality in choosing the number of layers, and the number of neurons per layer to get better optimal performance for ANN.

References

Al-Jbara, H. A. G., Kiah, M. L. B. M., & Jalab, H. A. (2012). Increased capacity of image based steganography using artificial neural network. **In *AIP Conf. Proc*** ,Vol. 1482, pp. 20-25.

Alsaif, K. I., & Salih, M. M. (2013). Contourlet Transformation for Text Hiding in HSV Color Image. ***International Journal of Computer Networks and Communications Security***, *1*(4), pp.132–139.

Al-Ataby, A., & Al-Naima, F. (2010). A modified high capacity image steganography technique based on wavelet transform. ***The International Arab Journal of Information Technology***, 7(4), pp 358-364.

Bender, W., Gruhl, D., Morimoto, N., & Lu, A. (1996). Techniques for data hiding. ***IBM systems journal***, *35*(3.4), pp. 313-336.

Bhattacharya, T., Dey, N., & Chaudhuri, S. R. (2012). A session based multiple image hiding technique using DWT and DCT. ***International Journal of Computer Applications***, *38* (5). pp. 18– 21.

Bhattacharyya, D., Bhaumik, A. K., Choi, M., & Kim, T. H. (2010). Directed graph pattern synthesis in LSB technique on video steganography. **In *Advances in Computer Science and Information Technology*** (pp. 61-69). Springer Berlin Heidelberg.

Cox, I., Miller, M., Bloom, J., Fridrich, J. & Kalker, T. (2007). ***Digital Watermarking and Steganography,*** Morgan Kaufmann publishers, Burlington, Second Edition, ISBN 978-0-12-372585-1.

Choudhury, D. R., Bhargava, P., & Reena, S. K. (2007). Use of Artificial Neural Networks for Predicting the Outcome of Cricket Tournaments. ***International Journal of Sports Science and Engineering***, *1*(2).pp 87-96.

Chow, T. W., & Cho, S. Y. (2007). ***Neural networks and computing: Learning algorithms and applications***. Vol. (7). Published by Imperial College Press, London.

Djebbar, F., Ayad, B., Meraim, K. A., & Hamam, H. (2012). Comparative study of digital audio steganography techniques. ***EURASIP Journal on Audio, Speech, and Music Processing***, *2012*(1), pp. 1-16. http://asmp.eurasipjournals.com/content/pdf/1687-4722-2012-25.pdf

George, J. P. (2012). *Development of efficient Biometric Recognition Algorithms Based on Fingerprint And Face*, PhD Thesis, Christ University.

Goresky, M., & Klapper, A. M. (2002). Fibonacci and Galois representations of feedback-with-carry shift registers. ***Information Theory, IEEE Transactions on***, *48*(11), pp. 2826-2836.

Hemalatha, S., Acharya, D. U., Renuka, A., & Kamath, P. R. (2013). A Secure Color Image Steganography in Transform Domain. ***International Journal on Cryptography and Information Security,*** *3* (1). PP. 17-24.

Hmood, A., Zaidan,A., Taqa, A. & Hamid, A. (2010). An Overview on Hiding Information Technique in image. ***Journal of Applied Sciences, Asian Network For Scientific Information***, 10 (18), pp. 2094-2100.

Haykin, S. (1999). ***Neural Networks: A Comprehensive Foundation***. Published by Pearson Prentice Hall, *Inc.*, *New Jersey*., Second Edition, ISBN 81-7808-300-0.

Heaton, J. (2008). ***Introduction to Neural Networks with Java***. Publisher: Heaton Research, USA, Second Edition, ISBN: 1604390085.

Islam, R., Naji, A. W., Zaidan, A. A., & Zaidan, B. B. (2010). New System for Secure Cover File of Hidden Data in the Image Page within Executable File Using Statistical Steganography Techniques. ***International Journal of Computer Science and Information Security (IJCSIS)***, *7* (1). pp. 273-279.

Johnson, N. F., Duric, Z., & Jajodia, S. (2001). ***Information Hiding: Steganography and Watermarking-Attacks and Countermeasures: Steganography and Watermarking: Attacks and Countermeasures*.** Kluwer Academic Publishers, USA, Vol. 1. Springer.

Jalab, H. A., Zaidan, A. A., & Zaidan, B. B. (2010). New Design for Information Hiding with in Steganography Using Distortion Techniques. ***International Journal of Engineering and Technology (IJET)***, ISSN, 8236, *2* (1). pp.72-77.

Jayasimman, L. & George, E. (2013). Classifying User Preferences of Web Learning System with Genetic Optimization. ***Journal of Engineering Research and Applications***, *3* (6) pp.1257-1261.

Kafri, N., & Suleiman, H. Y. (2009). Bit-4 of frequency domain-DCT steganography technique. **In** ***Networked Digital Technologies***, *2009. NDT'09. First International Conference on* , pp. 286-291. IEEE.

Kumar, K. S., Raja, K. B., Chhotaray, R. K., & Pattnaik, S. (2011). Performance comparison of robust steganography based on multiple transformation techniques. ***International Journal of Computer Technology and Applications***, 2 (4). pp. 1035-1047.

Kumar, S. & Multoo, S. (2011). Steganography based on contourlet Transform. ***International Journal of Computer Science and Information Security (IJCSIS)***, *9* (6), pp. 215-220.

Kumar, K. S., Raja, K. B., & Pattnaik, S. (2011). Hybrid domain in LSB steganography. ***International Journal of Computer Applications***. *19* (7). PP. 35-40.

Katzenbisser, S. & Petitcolas, F. (2000). ***Information Hiding Techniques for Steganography and Digital Watermarking***. Artech House publisher, Boston, ISBN 1-58053-035-4.

Kiang, M. Y. (2001). Extending the Kohonen self-organizing map networks for clustering analysis. ***Computational Statistics & Data Analysis***, *38* (2), pp. 161-180.

Kamau, G. M., Kimani, S., & Mwangi, W. (2012). An enhanced Least Significant Bit Steganographic Method for Information Hiding. ***Journal of Information Engineering and Applications***, 2(9). pp. 1-11.

Lippmann, R. P. (1987). An introduction to computing with neural nets. ***ASSP Magazine***, *IEEE*, *4*(2), pp. 4-22.

Lesly, L., & Roy, R. C. (2012). A Novel Adaptive Steganographic Technique using Kohonen Neural Network based on Integer Wavelet Transform. ***International Journal of Computer Applications***, *59* (17), pp.1-5.

Mandal, J. K. (2011). A Frequency Domain Steganography using Z Transform (FDSZT). ***International Workshop on Embedded Computing and Communication System (IWECC 2011)***, pp.1-4.

Mathkour, H., Assassa, G. M., Al Muharib, A., & Kiady, I. (2009). A novel approach for hiding messages in images. **In** ***Signal Acquisition and Processing***, *2009. ICSAP 2009. International Conference on* (pp. 89-93). IEEE.

Mahajan, R. & Kranthi, B. (2014). An Improved Image Steganography Technique Using Discrete Wavelet Transform. ***International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS)***, 9 (1), pp. 76-82.

Mistry, D. & Banerjee, A. (2013). Discrete Wavelet Transform Using Matlab. ***International Journal of Computer Engineering and Technology (IJCET)***, 4 (2), pp. 252-259.

Nain, V. & Bansal, N. (2014). Performance Upgradation of a Transform Domain based Steganography using Neural Logics. ***International Journal of Advanced Research in Computer Science & Technology (IJARCST 2014)***, *2* (1), pp. 150-153.

Narasimmalou, T., & Joseph, R. A. (2012, March). Discrete Wavelet Transform based steganography for transmitting images. **In** ***Advances in Engineering, Science and Management (ICAESM)***, *2012 International Conference on* (pp. 370-375). IEEE.

Naoum, R., Viktorov, O., Shihab, A., & Shaker, M. (2013). Image-to-mage Steganography Based on Discrete Cosine Transform. ***European Journal of Scientific Research***, *106* (4), pp. 512-522.

Naoum, R., Shaker, M., Mudhafar.J., & Shihab. A. (2014). Discrete Wavelet Transform for Image-to-Image Steganography. ***European Journal of Scientific Research***, *117* (1). pp.137-152.

Nitin, K., kirit, R., Avalik, R., Vijaysinh, J. & Ashish, N. (2014). A Novel Technique for Image Steganography Techniques Based on LSB and DCT Coefficients. ***International Journal for Scientific Research and Development (IJSRD)***, *1* (11). pp. 2479-2482.

Naoum, R., AlHamouz, S., Shihab, A. & Shaker, M. (2014). Image Steganography using Three Layers DCT and Artificial Neural Network. ***European Journal of Scientific Research***, *121* (3). pp. 226-240.

Naoum, R. (2011). *Lecture Notes, Artificial Neural Network* , Middle East University (MEU), Jordan.

Naoum, R. S., Abid, N. A., & Al-Sultani, Z. N. (2012). An Enhanced Resilient Backpropagation Artificial Neural Network for Intrusion Detection System. ***International Journal of Computer Science and Network Security IJCSNS***, *13(3),* pp. 98-104.

Odabas, M., Ergun, E. & Oner, F. (2013). Artificial Neural Network Approach For The Predicition of The Corn (ZEA MAYS L.) leaf Area. ***Bulgarian Journal of Agricultural Science***, *19* (4), pp. 766-796.

Petitcolas, F. A., Anderson, R. J., & Kuhn, M. G. (1999). Information Hiding- A Survey, *Proceedings of the IEEE*, *87*(7), pp.1062-1078.

Parah, S. A., Sheikh, J. A., & Bhat, G. M. (2012). Data hiding in intermediate significant bit planes, a high capacity blind steganographic technique. **In *Emerging Trends in Science, Engineering and Technology (INCOSET)***, *2012 International Conference on* (pp. 192-197). IEEE.

Parul., Manju., & Rohil, H. (2014). Optimized Image Steganography using Discrete Wavelet Transform (DWT), ***International Journal of Recent Development in Engineering and Technology (IJRDET)***, *2* (2). PP. 75-81.

Pejas, J., & Piegat, A. (2006). ***Enhanced methods in computer security, biometric and artificial intelligence systems*.** Kluwer Academic Publishers, Szczecin, Poland, Springer, ISBN 1-4020-7776-9.

Rodrigues, J. M., Rios, J. R., & Puech, W. (2004). SSB-4 System of Steganography using bit 4. In *5th* ***International Workshop on Image Analysis for Multimedia Interactive Services***.
https://hal.archives-ouvertes.fr/file/index/docid/108804/filename/D358.PDF

Rahman, M. (2013). A dwt, dct and svd based watermarking technique to protect the image piracy. ***International Journal of Managing Public Sector Information & Communication Technologies (IJMPICT)***, *4*(2). PP. 21-32.

Riedmiller, M., & Braun, H. (1993). A direct adaptive method for faster backpropagation learning: The RPROP algorithm. In *Neural Networks, 1993., IEEE International Conference on*, PP. 586-591. IEEE.

Stamp, M. (2006). ***Information Security Principles And Practice***. Published by John Wiley & Sons, Inc., Hoboken, New Jersey, 1st edition, ISBN-10 0-471-73848-4.

Sridevi, T., Swapna, K., & Kumar, V. V. (2011). Comparative Analysis of Normalization based Image Watermarking Techniques. ***International Journal of Computer Applications***, *27* (3), PP. 37-43.

Singh, S., & Siddiqui, T. J. (2012). Robust image steganography technique based on redundant discrete wavelet transform. In *Power, Control and Embedded Systems (ICPCES), 2012 2nd International Conference on*, pp.1-4. IEEE.

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

Scalero, R. S., & Tepedelenlioglu, N. (1992). A fast new algorithm for training feedforward neural networks. *Signal Processing, IEEE Transactions on*, 40 (1), PP. 202-210.

Sagar, G. V. R., Chalam, S. V., & Singh, M. K. (2011). Evolutionary algorithm for optimal connection weights in artificial neural networks. ***International Journal of Engineering (IJE)***, *5*(5), pp. 333-340.

Taqa, A., Zaidan, A. A., & Zaidan, B. B. (2009). New framework for high secure data hidden in the MPEG using AES encryption algorithm. ***International Journal of Computer and Electrical Engineering (IJCEE)***, *1*(5), pp. 566-571.

Umbaugh, S. E. (2010). ***Digital image processing and analysis: human and computer vision applications with CVIP tools***. CRC Press, Taylor and France Group, Second Edition.

Vani, B. G., & Prasad, E. V. (2013). High Secure Image Steganography based on Hopfield Chaotic Neural Network and Wavelet Transforms. ***International Journal of Computer Science and Network Security (IJCSNS)***, *13*(3). PP. 1-6.

Vijay, M., & Vignesh, V.(2014). Image Steganography Method Using Integer Wavelet Transform. *International Journal of Innovative Research in Science, Engineering and Technology, IEEE International Conference on Innovations in Engineering and Technology (ICIET'14)*. *3* (3), PP. 1207-1211.

Vijayakumar, M. S. (2011). Image Steganography based on Polynomial Functions. ***Journal of Global Research in Computer Science***, *2*(3). PP. 13-15.

Wu, M. & Liu, B. (2003). ***Multimedia Data Hiding***. Originally published by Springer-Verlag New York, USA, 1st edition, ISBN 978-1-4419-2994-5.

Yadav, R., Saini, R. & Kamaldeep. (2011). Cyclic Combination Method For Digital Image Steganography With Uniform Distribution Of Message. ***Advanced Computing: An International Journal (ACIJ)***, *2* (6). pp. 29-43.

Yan, W., & Weir, J. (2010). ***Fundamentals of Media Security***. Bookboon, Ventus Publishing Aps.
library.ku.ac.ke/resources/fundamentals-of-media-security.pdf

Zaidan, B. B., Zaidan, A. A., Taqa, A., & Othman, F. (2009). Stego-Image Vs Stego-Analysis System. ***International Journal of Computer and Electrical Engineering***, *1*(5). PP.1793-816.