Steganoflage:
A New Image Steganography Algorithm

Abbas Cheddad B.Sc./ M.Sc.

School of Computing & Intelligent Systems
Faculty of Computing & Engineering
University of Ulster

A thesis submitted in partial fulfilment of the requirements for the degree of
Doctor of Philosophy

September, 2009

Table of Contents

Table of Contents .... **ii**
List of Figures .... **v**
List of Tables .... **viii**
List of Acronyms .... **ix**
ACKNOWLEDGEMENTS .... **xi**
Abstract .... **xii**
Notes on access to contents .... **xiii**

Chapter 1: **Introduction** **1**
1.1 Motivations and Research Problem 2
1.2 Objectives of this thesis 3
1.3 Outline of this Thesis 4

Chapter 2: **Digital Image Steganography** **6**
2.1 Ancient Steganography 9
2.2 The Digital Era of Steganography 10
2.3 Steganography Applications 12
2.4 Steganography Methods 14
  2.4.1 Steganography exploiting the image format 16
  2.4.2 Steganography in the image spatial domain 18
  2.4.3 Steganography in the image frequency domain 24
  2.4.4 Adaptive steganography 30
2.5 Performance Analysis of Methods in the Literature with Recommendations 37
2.6 Steganalysis 45
2.7 Summary 51

Chapter 3: **Image Encryption Methods and Skin Tone Detection Algorithms** **53**
3.1 Image Encryption Methods 53
3.2 Skin Tone Detection Methods 61
  3.2.1 Orthogonal colour space (YC_bC_r) 64
  3.2.2 Log Opponent and HSV 66
  3.2.3 Basic N-rules RGB (NRGB) 68
  3.2.4 Other colour spaces 68

3.3 Summary 69

Chapter 4: **Steganoflage: Object-Oriented Image Steganography** **70**
4.1 Step 1: Payload Encryption (What to Embed?) 72
  4.1.1 A new image encryption algorithm 73
4.2. Step 2: Identifying Embedding Regions 76
4.3 Step 3: The Embedding 81
4.4 Summary 90

Chapter 5: **Implementation of Steganoflage** **92**
5.1 Development Environment 92
5.2 Architecture of *Steganoflage* 92
5.3 Bridging PHP to MATLAB 93
5.4 Applications of Steganoflage 97
  5.4.1 Combating digital forgery 98
    Motivations 98
    Methodology 100
  5.4.2 Multilayer security for patients' data storage and transmission 106
  5.4.3 Digital reconstruction of lost signals 107
5.5 Summary 111

Chapter 6: **Experimental Results** **112**
6.1 Security Analysis of the Image Encryption Method 112
  6.1.1 Key space analysis 112
  6.1.2 Key sensitivity analysis (malleability attack) 113
  6.1.3 Adjacent pixels analysis 113
  6.1.4 Randomness test / Distinguishing attack 116
    The Chi-square distribution 116
    Frequency test (monobit test) 117
    Runs test 120
    Cross-covariance sequence 121
  6.1.5 Differential analysis 123
  6.1.6 Other security issues 125
6.2 Evaluation of Skin Tone Detection Algorithm 131
6.3 Overall Robustness of *Steganoflage* 139

6.3.1 Robustness against intentional and passive attacks 140
6.3.2 Steganalysis and visual perceptibility 142
6.3.3 Limitations and merits 144
6.4 Summary 149

Chapter 7: **Conclusion and Future Work 150**
7.1 Summary 150
7.2 Relation to Other Work 153
7.2.1 Region-based image watermarking 154
7.2.2 Self-embedding 155
7.3 Future Work 155
7.3.1 Resilience to print-scan distortions (secure ID card) 156
7.3.2 Resilience to severe image lossy compression (iPhone) 156
7.3.3 Tamperproof CCTV surveillance 157
7.4 Conclusion 158

Appendix A: Bridging MATLAB to a Web Scripting Language 161
Appendix B: Image Encryption 162
Appendix C: Self-Embedding Examples 164
Appendix D: Key Sensitivity Analysis of the Image Encryption 168
Appendix E: Dark Skin-Tone Detection 170
Appendix F: Lossy Embedding with a Secret Angle 172

List of Figures

Figure 2.1: The different embodiment disciplines of Information Hiding 7
Figure 2.2: Media TV channels usually have their logos watermark.....8
Figure 2.3: Steganography versus watermarking 9
Figure 2.4: Cardan Grille 9
Figure 2.5: Concealment of Morse code, 1945 (Delahaye, 1996).....10
Figure 2.6: Fujitsu exploitation of steganography (BBC News, 2007) 13
Figure 2.7: Communication-theoretical view of a generic embedding process 14
Figure 2.8: Stego-image opened using Notepad.....17
Figure 2.9: Text insertion into EXIF header.....18
Figure 2.10: The effect of altering the LSBs up to the 4th bit plane 19
Figure 2.11: An implementation of steganography in the spatial domain 20
Figure 2.12: One byte representation 21
Figure 2.13: The system reported in Jung (Jung & Yoo, 2009) 23
Figure 2.14: Histogram distributions.....24
Figure 2.15: JPEG suggested Luminance Quantization Table 25
Figure 2.16: The modified Quantization Table (Li & Wang, 2007) 26
Figure 2.17: Data flow diagram of embedding in the frequency domain 27
Figure 2.18: DCT embedding artefacts.....28
Figure 2.19: Blocks of various complexity values (Hioki, 2002).....32
Figure 2.20: Images used to generate Table 2.3 39
Figure 2.21: Set A: stego-images of each software tool appearing in Table 2.3 39
Figure 2.22: Set B: stego-images of each software tool appearing in Table 2.3 40
Figure 2.23: Additional experiments on steganography software.....40
Figure 2.24: Steganalysis using visual inspection (Lin & Delp, 1999) 46
Figure 2.25: Histograms demonstrating the "*pair effect*".....47
Figure 2.26: Standard histograms may not reveal the structure of data 48
Figure 2.27: A test image for the RS steganalysis' performance 50

Figure 3.1: An example of chaotic map (Wu & Shih, 2006).....55
Figure 3. 2: The system provided by Hsu (Hsu et al., 2002) 65

Figure 4.1: The different components of the *Steganoflage* algorithm 72
Figure 4. 2: Block diagram of the proposed image encryption algorithm 74
Figure 4.3: Frequency distribution of the data 79
Figure 4.4: Skin tone segmentation using the proposed method 80
Figure 4.5: Test result of face segmentation in gray scale 81
Figure 4.6: The elliptical model formed by face features 84
Figure 4.7: Resistance to other deliberate image processing attacks 84
Figure 4.8: RBGC and PBC contrast in the graphical space.....87
Figure 4.9: Block diagram of the proposed steganography method 89
Figure 4.10: The proposed *Steganoflage* 90

Figure 5.1: Generic Architecture of *Steganoflage* 93

Figure 5.2: *Steganoflage* running with WampServer running in the background.............95
Figure 5.3: *Steganoflage*'s online user interface..95
Figure 5.4: Hyperlink created to view results on the browser 96
Figure 5.5: The generated results page "Report.html" 96
Figure 5.6: User agreement...97
Figure 5.7: *Steganoflage*'s offline application 97
Figure 5.8: Image fidelity in different binary representations..101
Figure 5.9: Self-embedding, Example 1 103
Figure 5.10: Self-embedding, Example 2 104
Figure 5.11: Visual distortion of Lena image...105
Figure 5.12: The advantage of the proposed algorithm 106
Figure 5.13: Embedding EPRs data in innocuous looking image.....................................107
Figure 5.14: Audio error concealment model using information hiding 108
Figure 5.15: JPEG compressed visualization of the audio signal.....................................108
Figure 5.16: Experiment on audio signal quasi-recovery...109
Figure 5.17: Error concealment in video streaming 110

Figure 6.1: Key sensitivity test...113
Figure 6.2: Correlation analysis of 5000 pairs of horizontal adjacent pixels.................114
Figure 6.3: Eradication of image statistics..115
Figure 6.4: The Chi-square χ^2 distribution of the original and encrypted signals.........117
Figure 6.5: Overcoming the frequency test 118
Figure 6.6: Performance of proposed method against AES...119
Figure 6.7: Monobit test on the encrypted images shown in Figure 6.6..........................120
Figure 6.8: Figure showing the randomness in natural images 121
Figure 6.9: Cross-covariance test for randomness...122
Figure 6.10: Goldhill- differential analysis..124
Figure 6.11: Lena- differential analysis 125
Figure 6.12: A 4x5 cropped plain patch from a natural image 126
Figure 6.13: CPA cryptanalysis attack 127
Figure 6.14: key sensitivity test for colour images 129
Figure 6.15: Noise attack 131
Figure 6.16: Skin detection in an arbitrary image..133
Figure 6.17: Performance analysis of skin tone detection on arbitrary images.............134
Figure 6.18: The first four frames from a standard testing video sequence 136
Figure 6.19: Performance comparison of different methods "Suzie.avi" 137
Figure 6.20: The first four frames from a Dell^{TM} video sequence 138
Figure 6.21: The first four frames and performance analysis on "Sharpness.wmv".......139
Figure 6.22: JPEG compression attack on the stego-image 140
Figure 6.23: Resistance to natural image processing attacks 141
Figure 6.24: Resistance to other deliberate image processing attacks 142
Figure 6. 25: Visual distortions..144
Figure 6.26: Embedding distortion...144
Figure 6.27: Using secret angle, 184^{o}, for the embedding and the extraction 145
Figure 6.28: Extracted hidden data using IWT 148

Figure 7.1: Simplified theoretical framework of a tamperproof surveillance system 158

Figure A.1: Parsing HTML code into MATLAB commands......161

Figure B.1: The main function to encrypt and decrypt digital images 163

Figure C.1: Doctored image-Victoria Memorial......164
Figure C.2: Doctored image-Duncreggan student village 165
Figure C.3: Doctored image-couple......166
Figure C.4: Doctored image- River Foyle 167

Figure D.1: Test-1......168
Figure D.2: Test-2......168
Figure D.3: Test-3......169

Figure E.1: The proposed skin-tone detection algorithm performance on dark skin 170

List of Tables

Table 2.1: Comparison of steganography, watermarking and cryptography 8
Table 2.2: Parameters of ABCDE (Hioki, 2002) 33
Table 2.3: Summary of performance of common software (Kharrazi et al., 2006) 38
Table 2.4: Comparison of different tools 41
Table 2.5: RS estimations (Fridrich et al., 2001a) 49

Table 5.1: Performance of different inverse halftoning algorithms 102
Table 5.2: Visual distortion of the cover using proposed algorithm 105

Table 6.1: Comparison of correlation analysis with recent methods 114
Table 6.2: Monobit test, the proposed method against AES 119
Table 6.3: Difference between encrypted images 124
Table 6.4: PSNR values of the different generated ciphers 129
Table 6.5: Comparison with different image encryption methods 130
Table 6.6: Comparison of computational complexity 135
Table 6.7: Distortion comparisons with other methods 148

Table 7.1: Drawbacks of current steganography 153
Table 7.2: Comparision of *Steganoflage* against Nikolaidis and Pitas work 155

List of Acronyms

CRYSTAL	CRYptography and encoding in the context of STeganographic Algorithms
RBGC	Reflected Binary Gray Code
PBC	Pure Binary Code
AES	Advanced Encryption Algorithm
PRNG	Pseudo-Random Number Generator
PHP	Hypertext Pre-processor
HTML	Hyper Text Markup Language
WYSIWYG	What You See Is What You Get
HVS	Human Visual System
.EXE	executable files
HSV	Hue, Saturation and Value components
NMI	neighbour mean interpolation
GIF	Graphics Interchange Format
JPEG	Joint Photographic Experts Group
PNG	Portable Network Graphics
LSB	Least Significant Bit
MSB	Most Significant Bit
EOF	End Of File
DCT	Discrete Cosine Transform
FT	Fourier Transform
DWT	Discrete Wavelet Transform
PM	Perceptual Masking
AS	Adaptive Steganography
EXIF	Extended File Information
BMP	Bit Map image
QT	Quantization Table
DFT	Discrete Fourier Transform
FFT	Fast Fourier Transform
PDF	Probability Density Function
iDFT	inverse Discrete Fourier Transform
PQ	Perturbed Quantization
STD	standard deviation
ABCDE	A Block Complexity based Data Embedding
MB1	model-based method 1
MB2	model-based method 2
BPCS	Bit Plane Complexity Segmentation
PSNR	Peak Signal-to-Noise Ratio
dB	Decibel
FCM	Fuzzy C-Means
CPA	Chosen-plaintext attack
CV	Computer Vision
ROI	Regions Of Interest
bpp	bit per pixel

χ^2	Chi-square
PSP	Preserving Statistical Properties
SVM	Support Vector Machine
QIM	Quantization Index Modulation
YASS	Yet Another Steganograhic System
MP	Markov process
DES	Data Encryption Standard
IDEA	International Data Encryption Algorithm
MD5	Message Digest 5
XOR	bitwise exclusive-OR
BCH	Bose-Chaudhuri Hochquenghem
HIS	Hue Intensity and Saturation
RGB	Red Green and Blue
YC_bC_r	Luminance (Y), chrominance blue (Cb) and chrominance red (Cr)
PCA	Principal Component Analysis
nRGB	normalized RGB
SCT	Spherical Coordinate Transform
SSC	Standard Skin Colour
LO	Log-Opponent
CDM	Colour Distance Map
SHA	Secure Hash Algorithm
FFT	Fast Fourier Transform
IrFFT	Irreversible Fast Fourier Transform
EM	Expectation Maximization
GMM	Gaussian Mixture Models
RDBMS	Relational Database Management System
EPRs	Electronic patient records
erfc	Complementary Error Function
NPCR	Number of Pixel Change Rate
ECB	Electronic Code Book
RCES	Random Control Encryption Subsystem
VOs	Video Objects
VOPs	Video Object Planes
IWT	Integer-to-integer Wavelet Transform
CCTV	Closed-Circuit TeleVision

ACKNOWLEDGEMENTS

I would like to express my gratitude to my supervisors Dr. Joan Condell, Dr. Kevin Curran and Prof. Paul Mc Kevitt for their guidance, kindness, motivations, suggestions and insight throughout this PhD research. Without them this thesis would not exist. I also thank the University of Ulster for awarding me the Vice Chancellor Research Studentship, VCRS. A particular thanks is due to Dr. Philip Morrow, Prof. Sally McClean, Mrs. Margaret Cooke in Research Graduate School and Mrs. Eileen Shannon in the Research Office. Their support and eagerness to provide the ideal research environment is absolute. Additionally, I thank the Intelligent Systems Research Centre, ISRC, staff, Prof. Martin McGinnity, Prof. Liam Maguire, Miss. Paula Sheerin and Mr. Peter Devine for providing their support and helping build the right atmosphere for this research to excel. I am grateful to all my friends at ISRC including but not limited to, Dr. Ammar Belatreche, Dr. Irfan Ghani, Ms. Sheila McCarthy, Ms. Julie Wall, Michael McBride and Dr. Cornelius Glackin for the friendship and the warm welcome to their group. I would also like to thank Mr. Ted Leath, Mr. Paddy McDonough, Mr. Bernard McGarry and Mr. Pat Kinsella for their help installing the required software and PC troubleshooting. I am indebted to all of the people above for their friendship and moral support which were among the things that kept me going. Thanks to Dr. John Macrae at the Office of Innovation for the assistance on grant applications. I would also like to thank RW Pierce™ Security Print Solutions, RAPTOR™ and CEM Systems™ for advice on commercialisation of aspects of this research. Last but not least, to my family, I offer my sincere thanks for their patient and unshakable faith in me.

Abstract

Steganography is the science that involves communicating secret data in an appropriate multimedia carrier, e.g., image, audio and video files. It comes under the assumption that if the feature is visible, the point of attack is evident, thus the goal here is always to conceal the very existence of the embedded data. It does not replace cryptography but rather boosts the security using its obscurity features. Steganography has various useful applications. However, like any other science it can be used for ill intentions. It has been propelled to the forefront of current security techniques by the remarkable growth in computational power, the increase in security awareness, e.g., individuals, groups, agencies, government and through intellectual pursuit. Steganography's ultimate objectives, which are undetectability, robustness, resistance to various image processing methods and compression, and capacity of the hidden data, are the main factors that separate it from related techniques such as watermarking and cryptography.

This thesis investigates current state-of-the-art methods and provides a new and efficient approach to digital image steganography. It also establishes a robust steganographic system called *Steganoflage*. *Steganoflage* advocates an object-oriented approach in which skin-tone detected areas in the image are selected for embedding where possible. The key objectives of this thesis are: 1) a new image encryption method tailored to digital images and steganography/watermarking, 2) a new, efficient and real-time skin-tone detection algorithm and 3) a new embedding method using the Reflected Binary Gray Code, RBGC, in the wavelet domain. Each of these components is tested against relevant performance measurements. The results are promising and point to the advocacy and coherence of the developed algorithm. A series of interesting applications are shown, i.e., combating digital forgery, multilayer security for patients' data storage and transmission and digital reconstruction of lost signals. Future work includes the integration of *Steganoflage* into some emerging technologies, such as the iPhone and CCTV, which require further enhancements in relation to severe compression tolerance and real-time execution.

Notes on access to contents

I hereby declare that with effect from the date on which the thesis is deposited in the Library of the University of Ulster, I permit the Librarian of the University to allow the thesis to be copied in whole or in part without reference to me on the understanding that such authority applies to the provision of single copies made for study purposes or for inclusion within the stock of another library. *This restriction does not apply to the British Library Thesis Service (which is permitted to copy the thesis on demand for loan or sale under the terms of a separate agreement) nor to the copying or publication of the title and abstract of the thesis*. IT IS A CONDITION OF USE OF THIS THESIS THAT ANYONE WHO CONSULTS IT MUST RECOGNISE THAT THE COPYRIGHT RESTS WITH THE AUTHOR AND THAT NO QUOTATION FROM THE THESIS AND NO INFORMATION DERIVED FROM IT MAY BE PUBLISHED UNLESS THE SOURCE IS PROPERLY ACKNOWLEDGED.

CHAPTER

ONE

Introduction

With advancements in digital communication technology and the growth of computer power and storage, the difficulties in ensuring individuals' privacy become increasingly challenging. The degrees to which individuals appreciate privacy differ from one person to another. Various methods have been investigated and developed to protect personal privacy. Encryption is probably the most obvious one, and then comes steganography. Encryption lends itself to noise and is generally observed while steganography is not observable.

Interest from the scientific community has escalated in the past few years in relation to steganography. This exhibits itself in the establishment of new dedicated conferences and books, increased funding from defence ministries, and the birth of various commercial companies. Needless to say that in a few countries, the burgeoning concern that leads to this generosity is as a result of the widespread paranoia of criminals and terrorists who may or may not use this method to communicate. Therefore, funding in those countries was biased towards counter-attacking steganography and paid little concern to enhancing the privacy of individuals. Unfortunately, the seed that sparked this fear was driven by a false alarm in an article in USA Today, a USA national newspaper, by Jack Kelley which had no evidence as will be shown in Chapter 2. Such an effort erupted into an open battle that has two unbalanced camps, one for creating steganography algorithms to backup the human need for privacy and another camp finding ways to defeat the newly developed methods, steganalysis.

This position is quite different from the attitude taken with cryptography for example. Governments invested huge money and resources to build an unbreakable encryption algorithm. This has never been the case with steganography.

Questions arise, such as whether child pornography exists inside seemingly innocent image or audio files? Are criminals transmitting their secret messages in such a way? Are anti-virus systems fooled each time by secret embedding? The answers are still not trivial. However, what is evident is that steganography can have some useful applications, and like other technologies, such as encryption, it can be misused.

This thesis advocates the importance of steganography not only for secure private communication but also for a range of other applications such as digital forgery detection and lost signals reconstruction.

1.1 Motivations and Research Problem

In recent years digital image-based steganography has established itself as an important discipline in signal processing. That is due in part to the strong interest from the research community. Unfortunately, given the high volume of the introduced techniques, the literature lacks a comprehensive review of these evolving methods. There are some initiatives in this regard, but most of them are out-dated surveys as discussed in Chapter 2.

All of the existing methods of steganography focus on the embedding strategy and give no consideration to the pre-processing stages, such as encryption, as they depend heavily on the conventional encryption algorithms which obviously are not tailored to steganography applications where flexibility, robustness and security are required. Andreas Westfeld, a steganography scholar at Dresden University, called upon researchers in the field to analyse the interaction between steganography and encryption, the crypto-stego interface (CRYSTAL, 2004).

Many of the current methods take for granted that resilience to noise, double compression, and other image processing manipulations are not required in the steganographic context. As such, in the warden passive attack scenario their hidden data will be destroyed or will not be retrievable.

Adaptive steganography aimed at identifying textural or quasi-textural areas for embedding the secret data runs into a few problems at the decoder side since its classification algorithms are not salient. In this thesis, skin-tone areas are the preferred choice for texture detection since the detection algorithm is robust and unique. Moreover, skin-tone areas always exhibit chrominance values residing along a middle range, therefore, the problem of underflow or overflow is overcome automatically.

In the process of searching for a good skin-tone detection algorithm, the various available techniques are proven to either be slow in execution and/or come with intolerable false alarms. Often, these algorithms neglect the fact that luminance can help improve their performance.

1.2 Objectives of this thesis

This thesis studies some innovative ways to enhance steganography in digital images. The objective of this work is to develop and validate a novel approach to provide performance enhancements over the steganography methods proposed in the literature.
The key objectives in this work are:

- A comprehensive and up-to-date survey on digital image steganography. The survey also provides analysis and critique (Cheddad et al., 2010).
- A new stream cipher for encrypting digital images that outperforms current solutions and that provides a balanced bit stream that mimics the white noise needed for steganographic applications. This resulted in a patent application filed and registered in the UK (Cheddad et al., 2008a).

- A new algorithm for skin-tone detection that is more accurate and faster than the techniques available in the literature. This resulted in another patent application filed and registered in the UK (Cheddad et al., 2008c).
- A paradigm of using the Reflected Binary Gray Code, RBGC, to enhance embedding the encrypted secret bits in the discrete wavelet domain which provides a model that meets both robustness as well as imperceptibility.

1.3 Outline of this Thesis

This thesis is organised into seven chapters. In Chapter 2, a survey of digital image steganography is presented. An attempt is made to differentiate between the three highly linked disciplines: steganography, cryptography and watermarking. The review starts by relating the work to other available surveys in the literature. This is followed by an update of the state-of-the-art development in the field of digital image steganography. Evaluation and critiques on each method are provided where possible. Some fundamental concepts pertaining to steganography are also presented including steganalysis. Digital image steganography in this review is grouped into three categories:

- Steganography in the Image Spatial Domain,
- Steganography in the Image Frequency Domain,
- Adaptive Steganography.

The categorization of steganographic algorithms into these three categories is unique to this work and there is no claim that it is a standard categorization. Adaptive methods can either be applied in the spatial or frequency domains. As such they are regarded as special cases of either of the former two categories. This work opts not to include image-format based steganography as it is a naïve implementation and extremely prone to detection.

Chapter 3 gives a review of image encryption methods such as Advanced Encryption Algorithm, AES, Pseudo-Random Number Generator, PRNG, stream and chaotic-based ciphers along with the major skin tone detection algorithms. This review is linked to part of the contributions.

Next, Chapter 4 examines in detail the theoretical aspects of the proposed system, which is an object oriented image steganography system that takes advantage of the developed image encryption and skin-tone algorithms. The chapter discusses in detail the processes and stages of the algorithm and the benefits it brings over the existing algorithms. It also answers the following three questions: What should be embedded? Where should it be embedded? How is it embedded?

Chapter 5 discusses the architecture and implementation of the proposed system along with a neat script that bridges MATLAB, which is a software for technical computing used to build the system, to web scripting languages that serves as the online interface of the algorithm. PHP, Hypertext Pre-processor, and HTML, Hyper Text Markup Language, are also incorporated into the system. Moreover, the chapter demonstrates the good side of using steganography where real-world problems can be solved practically. Applications of *Steganoflage* discussed are: combating digital forgery, multilayer security for patients' data storage and transmission and digital reconstruction of lost signals.

Qualitative and quantitative evaluations of *Steganoflage* compared with other related methods in the literature are given in Chapter 6. Analysis and results are reported that support the advocacy of the introduced algorithm. Since *Steganoflage* consists of different components, i.e., image encryption algorithm, skin-tone detection algorithm and wavelet embedding algorithm, Chapter 6 analyses each component and compares it to relevant related work.

Chapter 7 presents the conclusion. Important points in the thesis are summarised and future work covers some open issues which merit further consideration. Hence, the Chapter provides a summary of the thesis, relation to other work and future work.

CHAPTER

TWO

Digital Image Steganography

The concept of “What You See Is What You Get, WYSIWYG” which is encountered sometimes while printing images or other material is no longer precise and would not fool a steganographer as it does not always hold true. Images can be more than what can be seen with the Human Visual System, HVS, hence, they can convey more than merely 1000 words.

For decades people strove to develop innovative methods for secret communication. This chapter highlights some historical facts and attacks on methods, also known as steganalysis. A thorough history of steganography can be found in the literature (Johnson & Jajodia, 1998), (Judge, 2001) and (Provos & Honeyman, 2003).

Three techniques are interlinked, steganography, watermarking and cryptography. The first two are quite difficult to tease apart especially for those coming from different disciplines. Drawing a line between these techniques is both arbitrary and confusing (Wayner, 2002, p.2). Therefore, it is necessary to discuss briefly these techniques before a thorough review is provided. Figure 2.1 and Table 2.1 may eradicate such confusion. The work presented here revolves around steganography in digital images and does not discuss other types of steganography, such as linguistic or audio. Table 2.1 summarizes the differences and similarities between steganography, watermarking and cryptography.

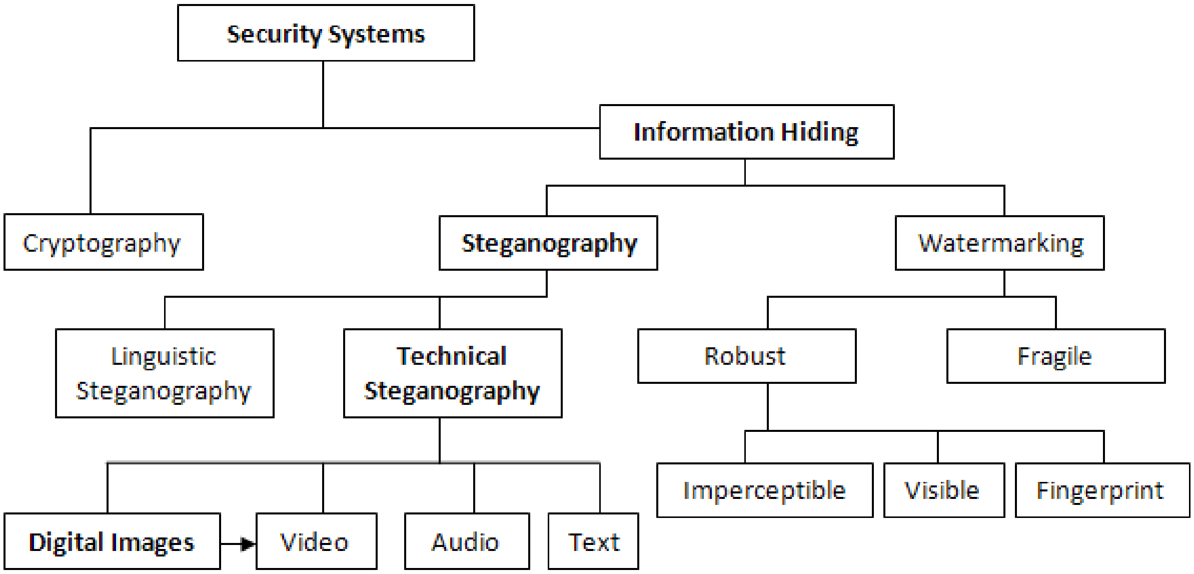


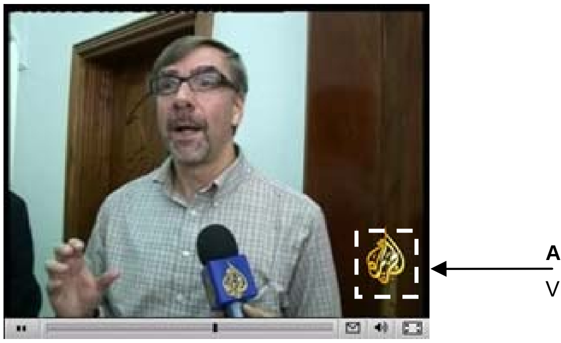
Figure 2.1: The different embodiment disciplines of Information Hiding. The arrow indicates an extension and bold face indicates the focus of this study

Figure 2.2 shows that media TV channels usually have their logos watermark for their broadcasting. Figure 2.3 demonstrates the main aims of steganography and watermarking, which are the exact extraction of the hidden data for steganography and the detection for watermarking. The figure also shows the attackers' main objectives, detection and destruction for steganography and watermarking, respectively. Figure 2.3 shows (top) aim of the embedder and (bottom) the aim of the attackers.

Intuitively, this work makes use of some nomenclature commonly used by steganography and watermarking communities. The term “cover image” is used throughout this thesis to describe the image designated to carry the embedded bits. An image with embedded data, payload, is described as “stego-image” while “steganalysis” or “attacks” refer to different image processing and statistical analysis approaches that aim to break steganography algorithms.

Table 2.1: Comparison of steganography, watermarking and cryptography

Criterion/*Method*	*Steganography*	*Watermarking*	*Cryptography*
Carrier	any digital media	mostly image/audio files	usually text based, with some extensions to image files
Secret data	payload	watermark	plain text
	no changes to the structure		changes the structure
Key	optional		necessary
Input files	at least two unless in self-embedding		one
Detection	blind	usually informative, i.e., original cover or watermark is needed for recovery	blind
Authentication	full retrieval of data	usually achieved by cross correlation	full retrieval of data
Objective	secrete communication	copyright preserving	data protection
Result	stego-file	watermarked-file	cipher-text
Concern	delectability/ capacity	robustness	robustness
Type of attacks	steganalysis	image processing	cryptanalysis
Visibility	never	sometimes, see Figure 2.2	always
Fails when	it is detected	it is removed/replaced	de-ciphered
Relation to cover	not necessarily related to the cover. The message is more important than the cover.	usually becomes an attribute of the cover image. The cover is more important than the message.	N/A
Flexibility	free to choose any suitable cover	cover choice is restricted	N/A
History	very ancient except its digital version	modern era	modern era



Aljazeera's Channel
Visible Watermark

Figure 2.2: Media TV channels usually have their logos watermark

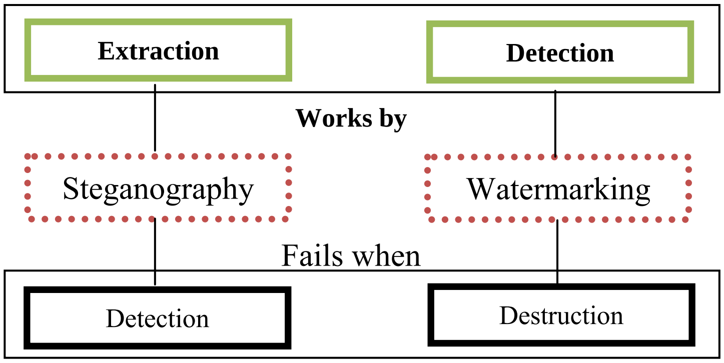


Figure 2.3: Steganography versus watermarking

2.1 Ancient Steganography

The word steganography is originally derived from Greek words which mean “Covered Writing”. It has been used in various forms for thousands of years. In the 5th century BC Histaiacus shaved a slave’s head, tattooed a message on his skull and the slave was dispatched with the message after his hair grew back (Johnson & Jajodia, 1998), (Judge, 2001), (Provos & Honeyman, 2003) and (Moulin & Koetter, 2005).

Five hundred years ago, the Italian mathematician Jérôme Cardan reinvented a Chinese ancient method of secret writing. The scenario goes as follows: a paper mask with holes is shared among two parties, this mask is placed over a blank paper and the sender writes his secret message through the holes then takes the mask off and fills the blanks so that the message appears as an innocuous text as shown in Figure 2.4. This is an illustration of the phenomenon. Note that the Grill has no fixed pattern: (left) the mask, (middle) the cover and (right) the secret message revealed. This method is credited to Cardan and is called Cardan Grille (Moulin & Koetter, 2005).

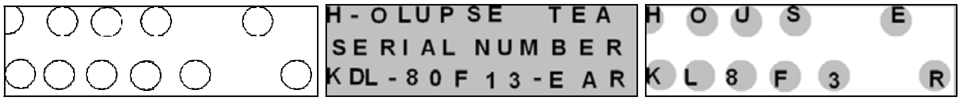


Figure 2.4: Cardan Grille. (Left) mask, (middle) cipher-text and (right) message revealed

It was also reported that, the Nazis invented several steganographic methods during World War II such as Microdots, and have reused invisible ink and null ciphers. As an example of the latter a message was sent by a Nazi spy that read: “Apparently neutral’s protest is thoroughly discounted and ignored. Isman hard hit. Blockade issue affects pretext for embargo on by-products, ejecting suets and vegetable oils.” Using the 2nd letter from each word the secret message is revealed: “Pershing sails from NY June 1” (Judge, 2001), (Lyu & Farid, 2006) and (Kahn, 1996).

In 1945, Morse code was concealed in a drawing, see Figure 2.5. The hidden information is encoded onto the stretch of grass alongside the river (Delahaye, 1996). The long grass denoted a line and the short grass denoted a point. The decoded message read: “Compliments of CPSA MA to our chief Col Harold R. Shaw on his visit to San Antonio May 11th 1945” (Delahaye, 1996).

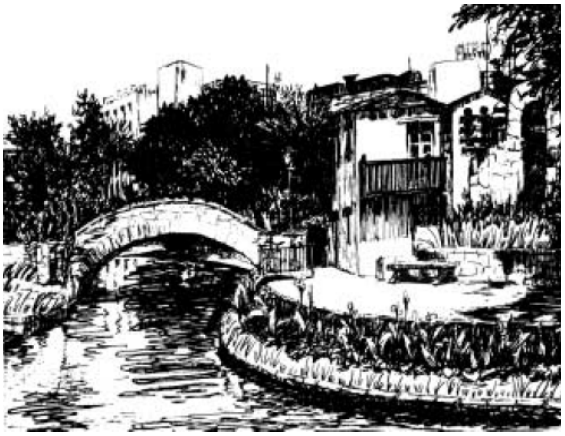


Figure 2.5: Concealment of Morse code, 1945 (Delahaye, 1996)

2.2 The Digital Era of Steganography

With the boost in computer power, the internet and with the development of digital signal processing, DSP, information theory and coding theory, steganography has gone “digital”. In the realm of this digital world steganography has created an atmosphere of corporate vigilance that has spawned various interesting applications, thus its continuing evolution is guaranteed.

Cyber-crime is believed to benefit from this digital revolution. Hence an immediate concern was shown on the possible use of steganography by criminals, following a report in USA TODAY[1]. Cyber-planning or the “digital menace” as Lieutenant Colonel Timothy L. Thomas defined it as being difficult to control (Thomas, 2003). Provos and Honeyman (Provos & Honeyman, 2003) scrutinized three million images from popular websites looking for any trace of steganography. They have not found a single hidden message. Despite the fact that they attributed several reasons to this failure it should be noted that steganography does not exist merely in still images. Embedding hidden messages in video and audio files is also possible. Examples exist in (Hosmer, 2006) for hiding data in music files, and even in a simpler form such as in Hyper Text Markup Language , HTML, executable files, .EXE, and Extensible Markup Language, XML (Hernandez-Castro et al., 2006). This shows that USA TODAY’s claim is not supported by strong evidence, if any, especially knowing that the writer of the above report resigned about two years later after editors determined that he had deceived them during the course of their investigation[2] (see also (McGill, 2005)).

Contemporary information hiding is due to (Simmons, 1984). Kurak and McHugh (Kurak & McHugh, 1992) discussed a method, which resembles embedding into the 4 LSBs, least significant bits. They examined image downgrading and contamination which is known now as image-based steganography. This Chapter’s focus is on the review of steganography in digital images. For a detailed survey on steganographic tools in other media from a forensic investigator’s perspective the reader is referred to (Hayati et al., 2007).

Section 2.3 briefly discusses the applications of steganography. Methods available in the literature are described in Section 2.4. The main discussions and comparisons focus on spatial domain methods, frequency domain methods and also adaptive methods in digital images. It will be shown that most of the steganographic algorithms discussed have been detected by steganalysis algorithms and thus a more robust approach needs

---

[1] USA TODAY: “*Researchers: No secret bin Laden messages on sites*”, (2001): <http://www.usatoday.com/tech/news/2001/10/17/bin-laden-site.htm#more>. Accessed on: 27-07-2009

[2] USA TODAY: “*Set to conduct independent probe*”, (2004); www.usatoday.com/news/2004-01-16-reporter_x.htm. Accessed on: 27-07-2009.

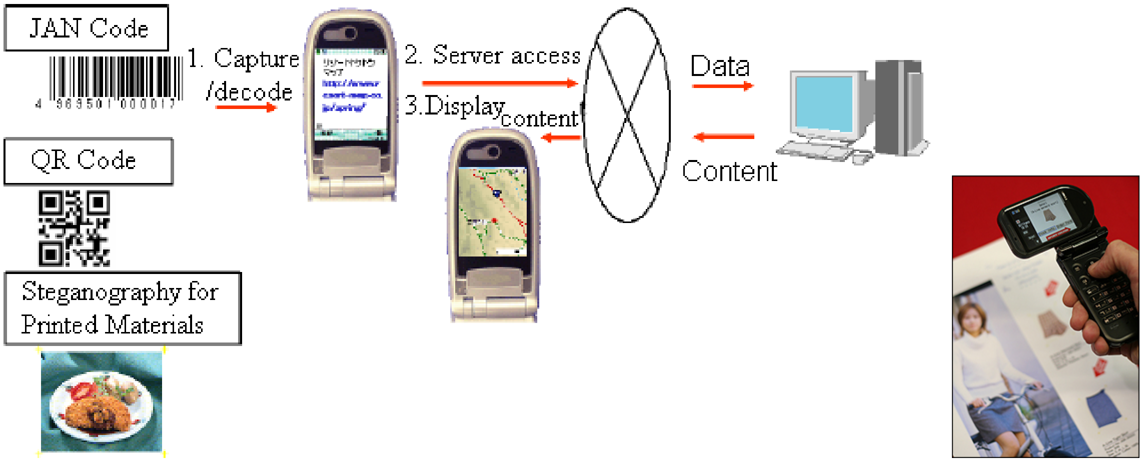
to be developed and investigated. Section 2.5 will give a brief analysis of the literature with some recommendations. Section 2.6 will briefly discuss the counterfeiting of steganography. A conclusion is provided in Section 2.7.

2.3 Steganography Applications

Steganography is employed in various useful applications, e.g., for human rights organizations, as encryption is prohibited in some countries (Frontline Defenders, 2003), copyright control of materials, enhancing robustness of image search engines and smart IDs, identity cards, where individuals' details are embedded in their photographs (Jain & Uludag, 2002). Other applications are video-audio synchronization, companies' safe circulation of secret data, TV broadcasting, TCP/IP packets, for instance a unique ID can be embedded into an image to analyze the network traffic of particular users (Johnson & Jajodia, 1998), and also checksum embedding (Chang et al., 2006a) and (Bender et al., 2000).

In (Petitcolas, 2000), the author demonstrated some contemporary applications, one of which was in Medical Imaging Systems where a separation was considered necessary for confidentiality between patients' image data or DNA sequences and their captions, e.g., physician, patient's name, address and other particulars. A link must be maintained between the image data and the personal information. Thus, embedding the patient's information in the image could be a useful safety measure and helps in solving such problems. Steganography would provide an ultimate guarantee of authentication that no other security tool may ensure. Miaou (Miaou et al., 2000) present an LSB embedding technique for electronic patient records based on bi-polar multiple-base data hiding. A pixel value difference between an original image and its JPEG version is taken to be a number conversion base. Nirinjan (Nirinjan & Anand, 1998) and Li (Li et al., 2007) also discuss patient data concealment in digital images.

Inspired by the notion that steganography can be embedded as part of the normal printing process, the Japanese firm Fujitsu is developing technology to encode data into a printed picture that is invisible to the human eye, but can be decoded by a mobile phone with a camera as exemplified in Figure 2.6 (BBC News, 2007).



(a) (b)

Figure 2.6: Fujitsu exploitation of steganography (BBC News, 2007). (a) shows a sketch representing the concept and (b) displays the application of deployment into a mobile phone

The process takes less than one second as the embedded data is merely 12 bytes. Hence, users will be able to use their cellular phones to capture encoded data. Fujitsu charges a small fee for the use of their decoding software which sits on the firm's own servers. The basic idea is to transform the image colour scheme prior to printing to its Hue, Saturation and Value components, HSV, then embed into the Hue domain to which human eyes are not sensitive. Mobile cameras can see the coded data and retrieve it. This application can be used for “doctor’s prescriptions, food wrappers, billboards, business cards and printed media such as magazines and pamphlets” (Frith, 2007), or to replace barcodes.

The confidence in the integrity of visual imagery has been ruined by contemporary digital technology (Farid, 2009). This has led to further research in the area of digital document forensics. Chapter 5 will discuss a proposed security scheme which protects scanned documents from forgery using self-embedding techniques. The method detects forgery and also allows legal or forensics experts to gain access to the original document despite the manipulation used.

2.4 Steganography Methods

This section attempts to give an overview of the most important steganographic techniques in digital images. The most popular image formats, non scientific images, used on the internet are Graphics Interchange Format, GIF, Joint Photographic Experts Group, JPEG, and to a lesser extent -Portable Network Graphics, PNG. Most of the techniques developed aimed to exploit the structures of these particular formats. There are some exceptions in the literature that use the Bitmap format, BMP, due to its simple data structure.

The process of embedding is defined as follows - a graphical representation is shown in Figure 2.7: Let *C* denote the cover carrier, i.e., image *A*, *M* the data to hide, $\hat{M}$ the extracted file, g_k the steganographic function and C' the stego-image. Let *K* represent an optional key, a seed used to encrypt the message or to generate a pseudorandom noise which can be set to {Ø}, the null set, for simplicity, and let *M* be the message to communicate, image *B*. *Em* is an acronym for embedding and *Ex* is an acronym for Extraction. Therefore, a complete steganographic system would be:

$$\mathrm{Em} : \mathrm{C} \oplus \mathrm{K} \oplus \mathrm{M} \rightarrow \mathrm{C}' \tag{2.1}$$

$$\therefore\ \mathrm{Ex}(\mathrm{Em}(c,k,m)) \approx m, \forall c \in \mathrm{C}, k \in \mathrm{K}, m \in \mathrm{M} \tag{2.2}$$

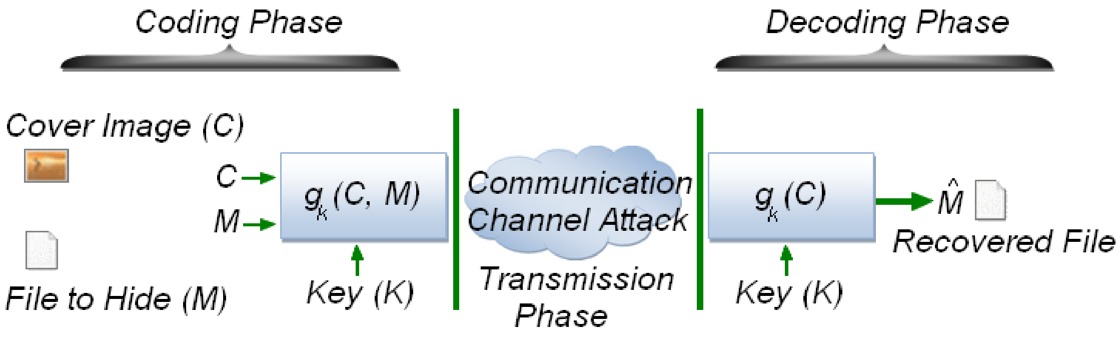


Figure 2.7: Communication-theoretical view of a generic embedding process

In this section some methods are briefly discussed which exploit image formats. Also some of the dominant techniques are detailed. The most popular survey available on steganographic techniques was published ten years ago (Johnson & Katzenbeisser, 2000). An evaluation of different spatial steganographic techniques applied especially to GIF images is also available (Bailey & Curran, 2006). The following contrasts the survey available later in this Chapter with the existing reviews in the literature.

In reference to the survey of Johnson (Johnson & Katzenbeisser, 2000):

The survey in this thesis is purely dedicated to steganography in image files, the most widespread research area. Johnson & Katzenbeisser discuss in: section 3.2.8, unused or reserved space in computer systems, section 3.3.2, hiding information in digital sound, section 3.3.3, echo hiding, section 3.6.1 encoding information in formatted text, section 3.7.1, mimics functions, section 3.7.2, automated generation of English texts.

Since the publication of the work (Johnson & Katzenbeisser, 2000), steganography has evolved dramatically. Therefore, an up-to-date survey is deemed necessary. In (Johnson & Katzenbeisser, 2000), the latest cited paper was published in 1999. This thesis' recommendations and method analysis can be distinguished from that of (Johnson & Katzenbeisser, 2000).

The classification, herein, of the techniques and that of Johnson et al. are different. Johnson et al. classify steganography techniques into: substitution systems, transform domain techniques, spread spectrum techniques, statistical methods, distortion techniques, and cover generation methods. The survey in (Johnson & Katzenbeisser, 2000) does not discuss the history of steganography nor its applications. The work in (Johnson & Katzenbeisser, 2000) has not included test images that can allow readers to visualize the concepts.

In reference to the survey of (Bailey & Curran, 2006):

The authors evaluate in their work some software that is applied in the spatial domain, mainly those supporting GIF formats (Bailey & Curran, 2006, p.62). However, they did not discuss or evaluate the frequency domain software/methods and did not criticize the core algorithms. In Bailey and Curran's work, published three years ago, the latest cited paper was published in 2001. They apply perceptual evaluation using a direct comparison between the original and stego-image files. Steganography assumes the unavailability of the original image. Their survey concludes the evaluation without recommendations or enhancements.

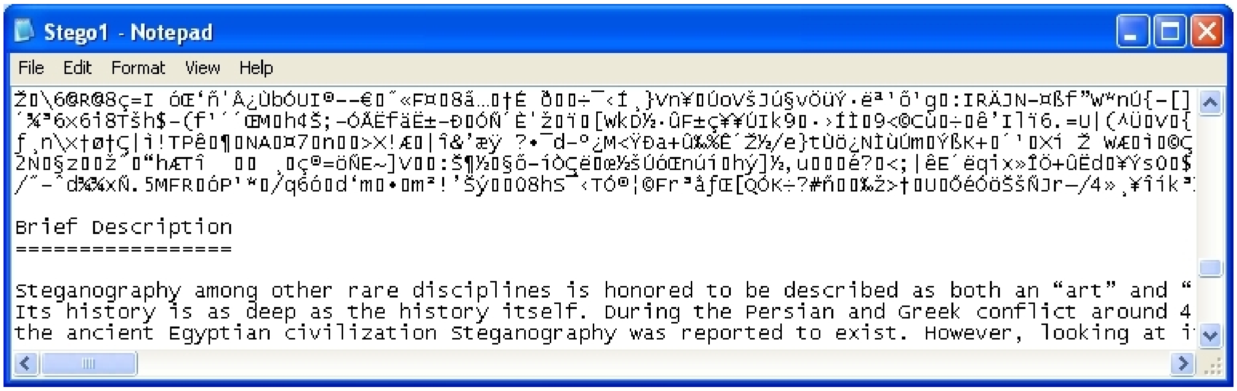
The remainder of this section provides a survey of the main steganographic methods. Section 2.4.1 discusses Spatial Domain techniques which generally use a direct Least Significant Bit, LSB, replacement method. Section 2.4.2 discusses the frequency domain based methods such as Discrete Cosine Transform, DCT, Fourier Transform, FT, and Discrete Wavelet Transform, DWT. Finally, Section 2.4.3 highlights the recent contribution in the domain which is termed Perceptual Masking, PM, or Adaptive Steganography, AS. The categorization of steganographic algorithms into the three categories, namely, spatial domain, frequency domain and adaptive methods, is unique to this work and there is no claim that it is a standard categorization. Adaptive methods can either be applied in the spatial or frequency domains and are thus regarded as special cases. Image-format based steganography techniques are not included here as they are naïve implementations and extremely prone to detection.

2.4.1 Steganography exploiting the image format

Steganography can be accomplished by simply feeding into a Windows operating system's command window the following code:

```
C:\> Copy Cover.jpg /b + Message.txt /b Stego.jpg
```

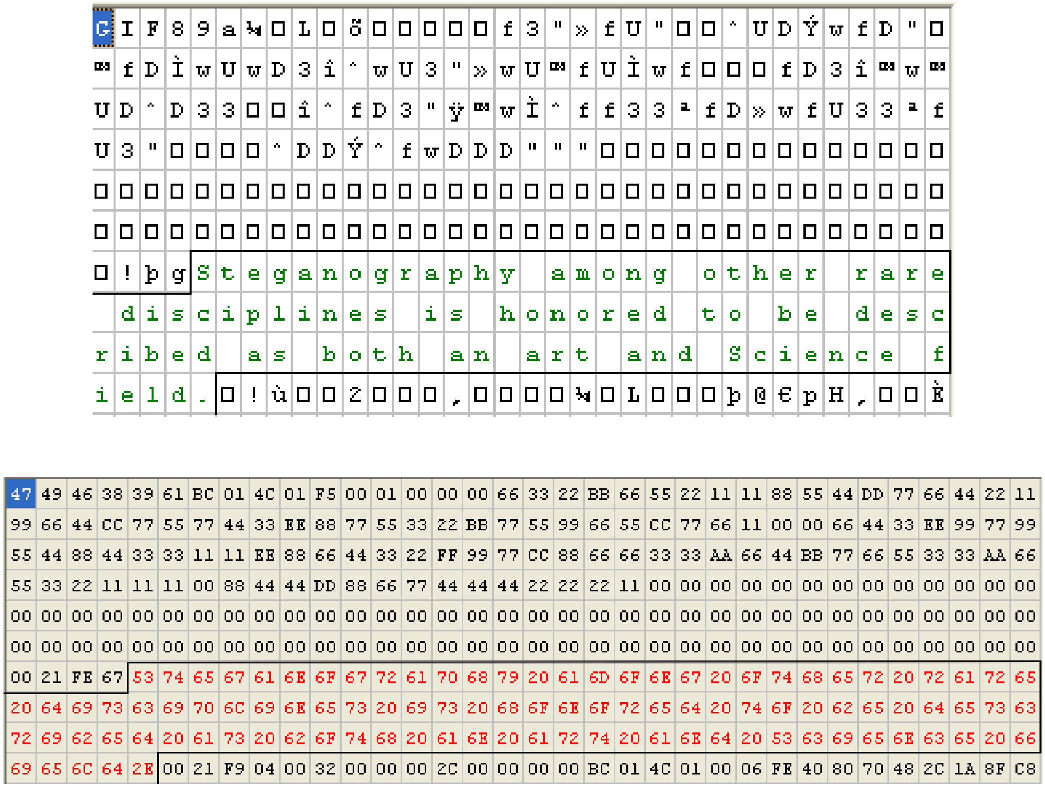
This code appends the secret message found in the text file 'Message.txt' into the JPEG image file 'Cover.jpg' and produces the stego-image 'Stego.jpg'. This attempts to abuse the recognition of EOF, End Of File. In other words, the message is packed and inserted after the EOF tag. When 'Stego.jpg' is viewed using any photo editing application, the latter will just display the picture ignoring anything coming after the EOF tag. However, when ‘Stego.jpg’ is opened in Notepad for example, the message reveals itself after displaying some data as shown in Figure 2.8. Note that the format of the inserted message remains intact. The embedded message does not impair the image quality. Neither image histograms nor visual perception can detect any difference between the two images due to the secret message being hidden after the EOF tag. Whilst this method is simple, a range of steganography software distributed online uses this method, Camouflage, JpegX, Data Stash (Online Software, n.d.). Unfortunately, this simple technique would not resist any kind of editing to the stego-image or any steganalysis attacks.



```
Ž□\6@R@8ç=I óŒ‘ñ'Â¿ÙbÓUI®--€□˜«F¤□8ã…□†É ð□□÷¯<Í¸}Vn¥□ÚoVšJú§VÖüÝ·ëª'õ'g□:IRÄJN–¤ßf”w*nÚ{-[]
´¾³6x6i8TŠh$-(f'´´ŒM□h4Š;-ÓÅËfäË±-Ð□ÓÑ´È'Ž□ï□[WkÐ½·ÛF±ç¥¥ÚIk9□·›ÎÌ□9<©CÙ□÷□ê'IlÏ6.=U|(^Ü□V□{
ƒ¸n\×†øţÇ|ì!TPê□¶□NA□¤7□n□□>X!Æ□|î&'æÿ ?•¯d-º¿M<ŸÐa+Û‰%Ê´Ž½/e}tÙö¿NÌùÚm□ŸßK+□´'□Xí Ž WÆ□ì□©Ç
ŽN□§z□□ž˝□“hÆTî  □□ ¸□ç®=öÑE~]V□□:Š¶½□§õ-íòÇë□œ½šÚóŒnúì□hý]½,u□□□é?□<;|êE´ëqix»ÎÖ+ûËd□¥Ÿs0□$
/˝-ˆd¾¾xÑ.5MFR□óP'*□/q6ó□d'm□•□m²!'Šý□□08hs¯‹TÓ®¦©Fr³åƒŒ[QÓK÷?#ñ□□‰ž>†□U□ÕéÓöŠšÑJr–/4»¸¥îik³

Brief Description
=================

Steganography among other rare disciplines is honored to be described as both an “art” and “
Its history is as deep as the history itself. During the Persian and Greek conflict around 4
the ancient Egyptian civilization Steganography was reported to exist. However, looking at i
```

Figure 2.8: Stego-image opened using Notepad. The above is image data followed by the inserted text

Another naïve implementation of steganography is to append hidden data into the image’s Extended File Information, EXIF. EXIF is a standard used by digital camera manufacturers to store information in the image file, such as, the make and model of a camera, the time the picture was taken and digitized, the resolution of the image, exposure time, and the focal length. This is metadata information about the image and its source located at the header of the file. Special agent Paul Alvarez (Alvarez, 2004) discussed the possibility of using such headers in digital evidence analysis to combat child pornography. Figure 2.9 depicts some text inserted into the comment field of a GIF image header. This method is not a reliable one as it suffers from the same drawbacks as that of the EOF method. Note that it is not always recommended to hide data directly without encrypting as in this example.



```
G I F 8 9 a ¼ □ L □ õ □ □ □ □ □ f 3 " » f U " □ □ ˆ U D Ý w f D " □
™ f D Ì w U w D 3 î ˆ w U 3 " » w U ™ f U Ì w f □ □ □ f D 3 î ™ w ™
U D ˆ D 3 3 □ □ î ˆ f D 3 " ÿ ™ w Ì ˆ f f 3 3 ª f D » w f U 3 3 ª f
U 3 " □ □ □ □ ˆ D D Ý ˆ f w D D D " " " □ □ □ □ □ □ □ □ □ □ □ □ □ □
□ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
□ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □
□ ! þ g S t e g a n o g r a p h y   a m o n g   o t h e r   r a r e
  d i s c i p l i n e s   i s   h o n o r e d   t o   b e   d e s c
r i b e d   a s   b o t h   a n   a r t   a n d   S c i e n c e   f
i e l d . □ ! ù □ □ 2 □ □ □ , □ □ □ □ ¼ □ L □ □ □ þ @ € p H , □ □ È
```

```
47 49 46 38 39 61 BC 01 4C 01 F5 00 01 00 00 00 66 33 22 BB 66 55 22 11 11 88 55 44 DD 77 66 44 22 11
99 66 44 CC 77 55 77 44 33 EE 88 77 55 33 22 BB 77 55 99 66 55 CC 77 66 11 00 00 66 44 33 EE 99 77 99
55 44 88 44 33 33 11 11 EE 88 66 44 33 22 FF 99 77 CC 88 66 66 33 33 AA 66 44 BB 77 66 55 33 33 AA 66
55 33 22 11 11 11 00 88 44 44 DD 88 66 77 44 44 44 22 22 22 11 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 21 FE 67 53 74 65 67 61 6E 6F 67 72 61 70 68 79 20 61 6D 6F 6E 67 20 6F 74 68 65 72 20 72 61 72 65
20 64 69 73 63 69 70 6C 69 6E 65 73 20 69 73 20 68 6F 6E 6F 72 65 64 20 74 6F 20 62 65 20 64 65 73 63
72 69 62 65 64 20 61 73 20 62 6F 74 68 20 61 6E 20 61 72 74 20 61 6E 64 20 53 63 69 65 6E 63 65 20 66
69 65 6C 64 2E 00 21 F9 04 00 32 00 00 00 2C 00 00 00 00 BC 01 4C 01 00 06 FE 40 80 70 48 2C 1A 8F C8
```

Figure 2.9: Text insertion into EXIF header, (top) the inserted text string highlighted in a box and (bottom) its corresponding hexadecimal chunk

2.4.2 Steganography in the image spatial domain

In spatial domain methods a steganographer modifies the secret data and the cover medium, which involves encoding at the level of the LSBs. This method, although simpler, has a larger impact compared to the other two types of methods (Alvarez, 2004). A general framework showing the underlying concept is highlighted in Figure 2.10.

A practical example of embedding from the 1st LSB to the 4th LSB is illustrated in Figure 2.11. It can be seen that embedding in the 4th LSB generates more visual distortion to the cover image as the hidden information is seen as “non-natural”.

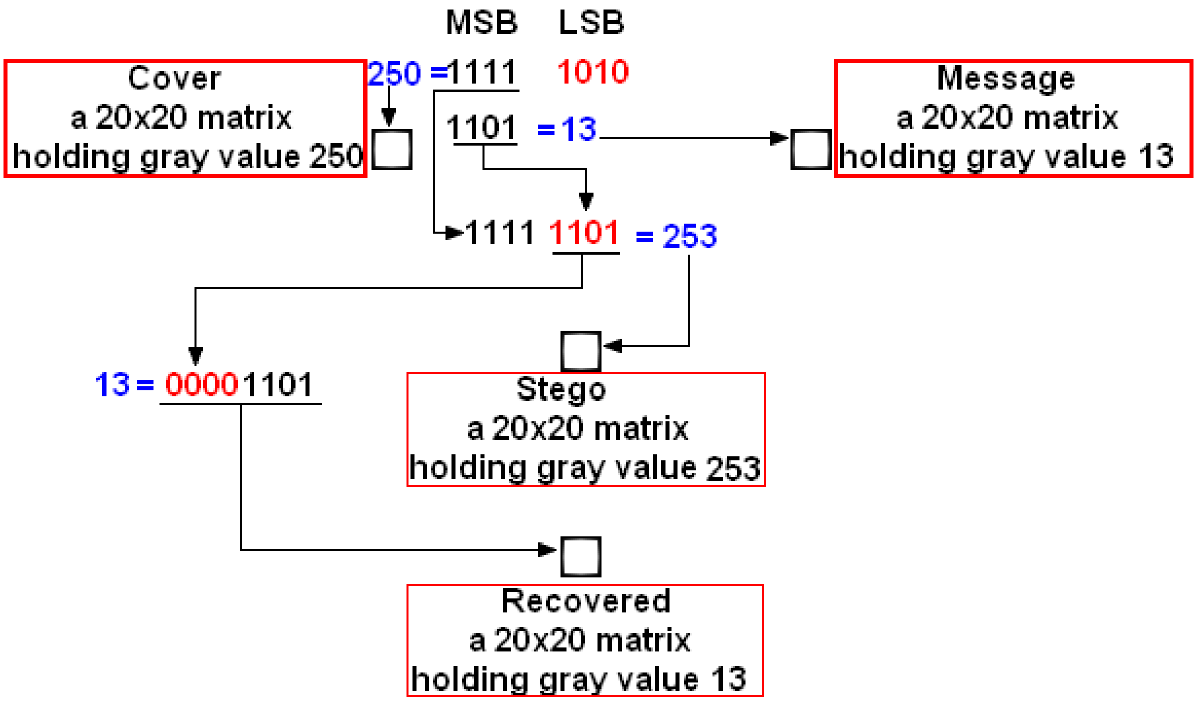
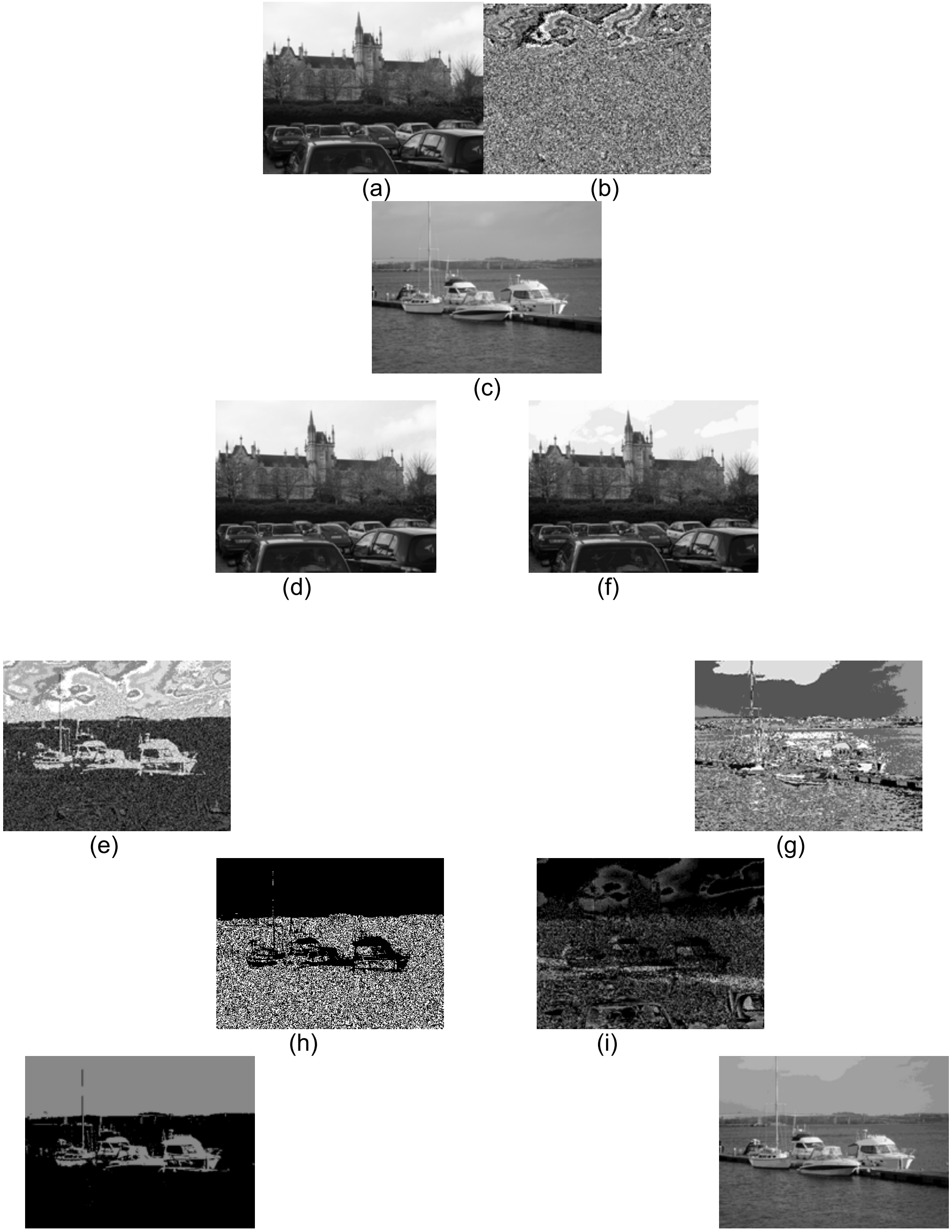


Figure 2.10: The effect of altering the LSBs up to the 4th bit plane

It is apparent to an observer that

Figure 2.11 concludes that there is a trade-off between the payload and the cover image distortion. However the payload is analogous with respect to the recovered embedded image when embedding in up to the 1st, 2nd, 3rd, or 4th LSB. For instance,

Figure 2.11 (k), recovered from embedding into four LSBs, is a good estimate of the hidden image, Figure 2.11 (c), but produces noticeable artefacts, Figure 2.11(f). On the other hand, Figure 2.11(j), recovered from embedding into 1st LSB, trades bad quality with an almost identical carrier to the original, compare Figure 2.11 (d) with Figure 2.11(a).


(a) (b)

(c)

(d) (f)

(e) (g)

(h) (i)

(j) (k)

Figure 2.11: An implementation of steganography in the spatial domain. (a) The cover carrier - University of Ulster , (b) 1st-4th LSBs of *(a)* with the contrast being enhanced for better visualization, (c) The image to hide - Derry's river- , (d) Stego-image 1st LSBs replaced with 1st MSBs, most significant bits, of *(c)*, (e) LSBs of *(d)*, (f) Stego-image 1st-4th LSBs replaced with 1st-4th MSBs of *(c)*, (g) LSBs of *(f)* , (h) Difference between *(a)* and *(d)*, (i) Difference between *(a)* and *(f)*, (j) Hidden image extracted from *(d)*, (k) Hidden image extracted from *(f)*

Figure 2.12 shows another trade-off between bit level and embedding distortion. It is clear that choosing the correct index for embedding is very crucial. This intricacy is less severe when using the RBGC since it produces seemingly disordered decimal-to-binary representation.

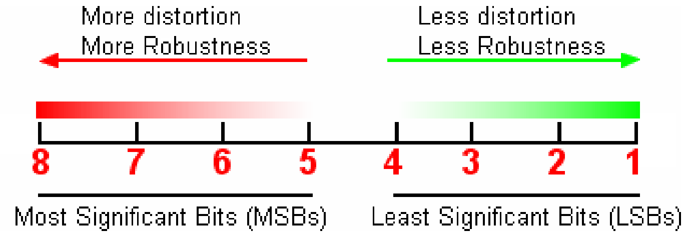


Figure 2.12: One byte representation with the conventional integer to binary conversion

Potdar (Potdar et al., 2005b) used a spatial domain technique in producing a fingerprinted secret sharing steganography for robustness against image cropping attacks. Their paper addressed the issue of image cropping effects rather than proposing an embedding technique. The logic behind their proposed work was to divide the cover image into sub-images and compress and encrypt the secret data. The resulting data was then sub-divided in turn and embedded into those image portions. To recover the data, a Lagrange Interpolating Polynomial was applied along with an encryption algorithm. The computational load was high, but their algorithm parameters, namely the number of sub-images, *n*, and the threshold value, *k*, were not set to optimal values leaving the reader to guess the values. Notice also that if *n* is set to 32, for example, that means 32 public keys are needed along with 32 persons and 32 sub-images, which turns out to be impractical. Moreover, data redundancy that they intended to eliminate occurs in the stego-image.

Shirali-Shahreza (Shirali-Shahreza & Shirali-Shahreza, 2006) exploited Arabic and Persian alphabet punctuations to hide messages. While their method is not related to the LSB approach, it falls into the spatial domain if the text is treated as an image. Unlike English language, which has only two letters with dots in their lower case format, namely “i” and “j”, the Persian language is rich in that 18 out of 32 alphabet letters have dots.

The secret message is binarized and those 18 letters' dots are modified according to the values in the binary file.

Colour palette based steganography exploits the smooth ramp transition in colours as indicated in the colour palette. The LSBs are modified based on their positions in the palette index. Johnson and Jajodia (Johnson & Jajodia, 1998) were in favour of using BMP, 24-bit, instead of JPEG images. Their next-best choice was GIF files, 256-color. BMP as well as GIF based steganography apply LSB techniques, while their resistance to statistical counter attacks and compression are reported to be weak (Provos & Honeyman, 2003), (Lin & Delp, 1999), (Chang et al., 2006b), (Hwang et al., 2001) and (Kong et al., 2005). BMP files are bigger compared to other formats which render them improper for network transmissions. JPEG images however, initially were avoided because of their compression algorithm which does not support a direct LSB embedding into the spatial domain. In (Fridrich et al., 2002), the authors claimed that changes as small as flipping the LSB of one pixel in a JPEG image can be reliably detected. The experiments on the DCT coefficients showed promising results and redirected researchers' attention towards this type of image. In fact acting at the level of DCT makes steganography more robust and less prone to statistical attacks.

Jung (Jung & Yoo, 2009) down-sampled an input image to ½ of its size and then used a modified interpolation method, termed the neighbour mean interpolation, NMI, to up-sample the result back to its original dimensions ready for embedding. For the embedding process the up-sampled image was divided into 2x2 non-overlapping blocks as shown in Figure 2.13. Potential problems with this method are:

- the impossibility of recovering the secret bits without errors, owing to the use of $\log_2$, base 2 logarithm, which is also used in the extraction that produces floating point values that can be displayed in different precision in different machines, e.g. rounding issue.

- since in the 2x2 blocks, the leading value, i.e., block(1,1), is left unaltered, thus this leads to the destruction of the naturally strong correlation between adjacent pixels which advertises a non-natural process involvement.
- this method resembles to a certain extent the pixel-value differencing for reversible data embedding method which is proven to be prone to histogram analysis attacks (Zhang & Wang, 2004). Also Tian (Tian, 2003) commented on the method's vulnerability.

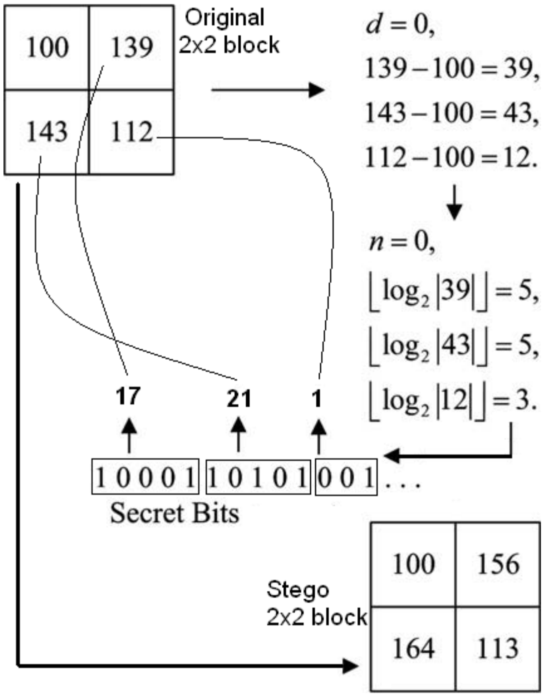


Figure 2.13: The system reported in Jung (Jung & Yoo, 2009)

Histogram-based data hiding is another commonly used data hiding scheme. Li (Li et al., 2009) propose lossless data hiding using the difference value of adjacent pixels. It is classified under '±1' data embedding algorithms. It exploits the correlation between adjacent pixels that more likely results in a compact histogram that is characterized by a normal Gaussian distribution, see Figure 2.14. Instead of considering the whole image, Piyu Tsai (Tsai et al., 2009) divides the image into blocks of 5x5 where the residual image is calculated using linear prediction, another term for adjacent pixels' difference.

The secret data is then embedded into the residual values, followed by block reconstruction.

Such schemes have the advantage of recovering the original cover image from the stego-image. While this preservation can be required in certain applications such as medical imaging, in general steganography is not concerned with this recovery. The hiding capacity is restricted in these methods, besides the '±1' embedding strategy can be detected, see, for example (Cancelli et al., 2008).

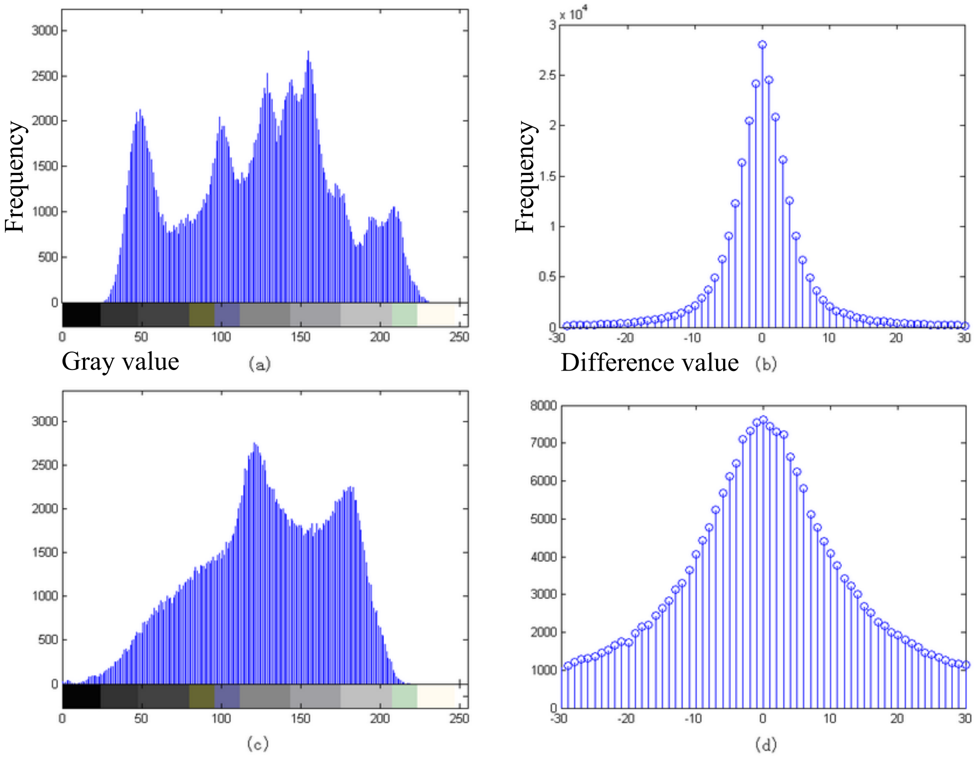


Figure 2.14: Histogram distributions. (a) histogram of Lena, a standard test image (b) difference histogram of Lena, (c) histogram of Baboon, (d) difference histogram of Baboon (Tsai et al., 2009)

2.4.3 Steganography in the image frequency domain

New algorithms keep emerging prompted by the performance of their ancestors, spatial domain methods, by the rapid development of information technology and by the need for an enhanced security system. The discovery of the LSB embedding mechanism is actually a big achievement. Although it is perfect in not deceiving the HVS, its weak resistance to attacks left researchers wondering where to apply it next until they successfully applied it within the frequency domain. The description of the two-dimensional DCT for an input image *F* and an output image *T* is calculated as:

$$T_{pq} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} F_{mn} \cos\frac{\pi(2m+1)p}{2M} \cos\frac{\pi(2n+1)q}{2N}, \qquad (2.3)$$

where

$0 \le p \le M-1$

$0 \le q \le N-1$

and

$$\alpha_p = \begin{cases} 1/\sqrt{M}, p = 0 \\ \sqrt{2/M}, 1 \le p \le M-1 \end{cases} \qquad \alpha_q = \begin{cases} 1/\sqrt{N}, q = 0 \\ \sqrt{2/N}, 1 \le q \le N-1 \end{cases}$$

where *M*, *N* are the dimensions of the input image while *m*, *n* are variables ranging from 0 to *M-1* and 0 to *N-1* respectively.

DCT is used extensively with video and image compression e.g. JPEG lossy compression. Each block DCT coefficient obtained from Equation (2.3) is quantized using a specific Quantization Table, QT. This matrix shown in Figure 2.15 is suggested in the Annex of the JPEG standard. Note that some camera manufacturers have their own built-in QT and they do not necessarily conform to the standard JPEG table. The value 16, in bold-face in Figure 2.15, represents the DC coefficient and the other values are the AC coefficients. The logic behind choosing a table with such values is based on extensive experimentation that tried to balance the trade-off between image compression and quality factors. The HVS dictates the ratios between values in the QT.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Figure 2.15: JPEG suggested Luminance Quantization Table. See Chapter 3 for further detail on colour spaces

The aim of quantization is to loosen up the tightened precision produced by DCT while retaining the valuable information descriptors. The quantization step is specified by:

$$f'(\omega_x,\omega_y)=\left\lfloor \frac{f'(\omega_x,\omega_y)}{\Gamma(\omega_x,\omega_y)}+\frac{1}{2}\right\rfloor,\ \omega_x,\omega_y \in 0,1,\ldots,7 \qquad (2.4)$$

where, x and y are the image coordinates, $f'(\omega_x,\omega_y)$ denotes the result function, $f(\omega_x,\omega_y)$ is an 8x8 non-overlapping intensity image block and $\lfloor\cdot\rfloor$ a floor rounding operator. $\Gamma(\omega_x,\omega_y)$ represents a quantization step which, in relationship to JPEG quality, is given by:

$$\Gamma(\omega_x,\omega_y)=\begin{cases}\max\left(\left\lfloor \frac{200-2Q}{100}QT(\omega_x,\omega_y)+\frac{1}{2}\right\rfloor,1\right) & ,\ 50\le Q\le 100\\ \left\lfloor \frac{50}{Q}QT(\omega_x,\omega_y)+\frac{1}{2}\right\rfloor & ,\ 0\le Q\le 50\end{cases} \qquad (2.5)$$

where, $QT(\omega_x,\omega_y)$ is the quantization table depicted in Figure 2.15 and Q is a quality factor. JPEG compression then applies entropy coding such as the Huffman algorithm to compress the resulting $\Gamma(\omega_x,\omega_y)$. Most of the redundant data and noise are lost in this stage hence the name lossy compression. For more details on JPEG compression the reader is directed to Popescu's work (Popescu, 2005).

The above scenario is a discrete theory independent of steganography. Li and Wang (Li & Wang, 2007) presented a steganographic method that modifies the QT and inserts the hidden bits in the middle frequency coefficients. Their modified QT is shown in Figure 2.16. The new version of the *QT* gives 36 coefficients in each 8x8 block within which to embed the secret data, yielding a reasonable payload. Their work was motivated by a prior published work (Chang et al., 2002).

8	1	1	1	1	1	1	1
1	1	1	1	1	1	1	55
1	1	1	1	1	1	69	56
1	1	1	1	1	87	80	62
1	1	1	1	68	109	103	77
1	1	1	64	81	104	113	92
1	1	78	87	103	121	120	101
1	92	95	98	112	100	103	99

Figure 2.16: The modified Quantization Table (Li & Wang, 2007)

Steganography based on DCT, JPEG compression, follows steps as shown in Figure 2.17.

Most of the techniques here use JPEG images within which to embed the data. JPEG compression uses the DCT to transform successive sub-image blocks, 8x8 pixels, into 64 DCT coefficients. Data is inserted into these coefficients' insignificant bits, however, altering any single coefficient would affect the entire 64 block pixels (Fard et al., 2006). As the change is operating on the frequency domain instead of the spatial domain there will be less visible change in the cover image given those coefficients are handled with care (Hashad et al., 2005).

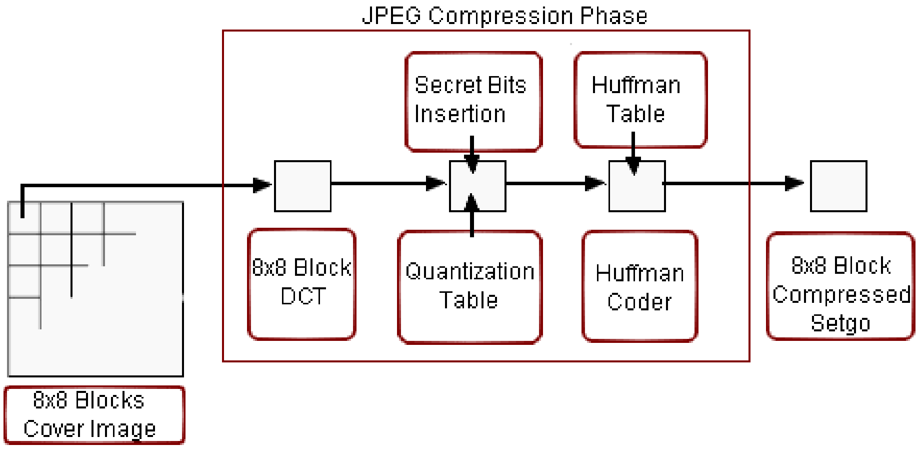
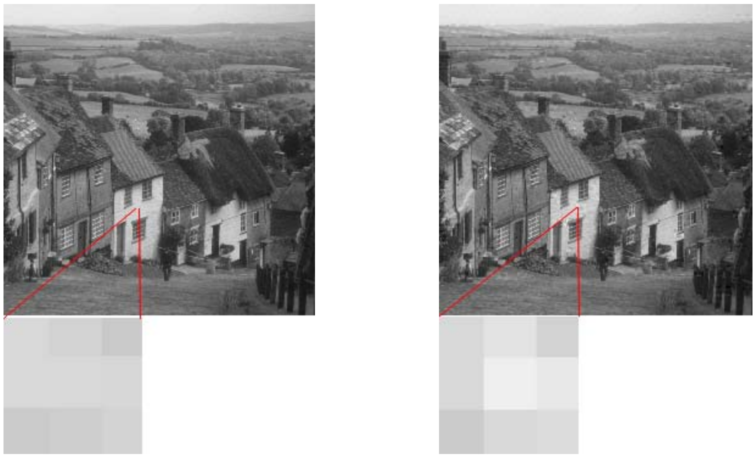


Figure 2.17: Data flow diagram of embedding in the frequency domain

According to Raja (Raja et al., 2005) Fast Fourier Transform, FFT, methods introduce round off errors, thus are not suitable for hidden communication. However, Johnson and Jajodia (Johnson & Jajodia, 1998), included these methods among the used transformations in steganography and another author utilised the 2D Discrete Fourier Transform, DFT, to generate Fourier based steganography in movies (McKeon, 2007).

Choosing which values in the 8x8 DCT coefficients block are altered is very important as changing one value will affect the whole 8x8 block in the image. Figure 2.18 shows a poor implementation of such a method in which careful consideration was not given to the sensitivity of DCT coefficients resulting in some artefacts becoming noticeable.



Original 3x3 pixels block zoomed Stego-image 3x3 pixels block zoomed

Figure 2.18: DCT embedding artefacts. (Left) patch from an original image (right) patch from the stego-image

The JSteg algorithm was among the first algorithms to use JPEG images. Although the algorithm stood strongly against visual attacks, it was found that examining the statistical distribution of the DCT coefficients shows the existence of hidden data (Provos & Honeyman, 2003). JSteg is easily detected using the χ^2-test (Westfeld & Pfitzmann, 1999). Moreover, since the DCT coefficients need to be treated with sensitive care and intelligence the JSteg algorithm leaves a significant statistical signature. In his book, Wayner, stated that the coefficients in JPEG compression normally fall along a bell curve and the hidden information embedded by JSteg distorts this (Wayner, 2002). Manikopoulos (Manikopoulos et al., 2002) discussed an algorithm that utilises the Probability Density Function, PDF, to generate discriminator features fed into a neural network system which detects hidden data in this domain.

OutGuess is a better alternative as it uses a pseudo-random-number generator to select DCT coefficients (Provos & Honeyman, 2003). The χ^2-test does not detect data that is randomly distributed. The developer of OutGuess suggests a counter attack against his algorithm. Provos (Provos & Honeyman, 2003), (Provos, 2001) and (Provos & Honeyman, 2001) suggest applying an extended version of the χ^2 -test to select Pseudo-randomly embedded messages in JPEG images.

Andreas Westfeld bases his “F5” algorithm (Westfeld, 2001) on subtraction and matrix encoding, also known as syndrome coding. F5 embeds only into non-zero AC DCT coefficients by decreasing the absolute value of the coefficient by *1*. A shrinkage occurs, as described in (Fridrich et al., 2007), when the same bit has to be re-embedded in case the original coefficient is either ‘1’ or ‘-1’ as at the decoding phase all zero coefficients will be skipped whether they are modified or not. Neither the χ^2-test nor its extended version could break this solid algorithm. Unfortunately, F5 did not survive attacks for too long. Fridrich (Fridrich et al., 2002) proposed steganalysis method, by exploiting the natural distribution of DCT coefficients, which does detect F5 contents, disrupting its survival.

Another trend related to the above quantization table modification, Figure 2.16, is the so-called Perturbed Quantization, PQ, (Fridrich et al., 2005), which aims to achieve high efficiency, with minimal distortion, rather than a large capacity. Each coefficient in the DCT block is assigned a scalar value that corresponds to how much impact it would have on the carrier image, and then a steganographer can set a selection rule to filter out the “well behaved” coefficients, thus giving the algorithm less payload but high imperceptibility.

Although the above frequency domain techniques live in the DCT coefficients, they fail in retrieving the embedded data if the stego-image is re-compressed.

As for steganography in the DWT, the reader is directed to some examples in the literature (Chen, 2007), (Potdar et al., 2005a) and (Verma et al., 2005). Abdulaziz and Pang (Abdulaziz & Pang, 2000) use vector quantization called Linde-Buzo-Gray, LBG, coupled with Block codes known as BCH code and 1-Stage discrete Haar Wavelet transforms. They reaffirm that modifying data using a wavelet transformation preserves good quality with little perceptual artefacts. The DWT-based embedding technique is still in its infancy. Paulson (Paulson, 2006) reports that a group of scientists at Iowa State University are focusing on the development of an innovative application which they call "Artificial Neural Network Technology for steganography, ANNTS," aimed at detecting all present steganography techniques including DCT, DWT and DFT. The Inverse Discrete Fourier Transform, iDFT, encompasses round-off error which renders DFT improper for steganography applications.

Abdelwahab and Hassan (Abdelwahab & Hassan, 2008) propose a data hiding technique in the DWT domain. Both secret and cover images are decomposed using DWT, 1st level each of which is divided into disjoint 4x4 blocks. Blocks of the secret image fit into the cover blocks to determine the best match. Error blocks are then generated and embedded into coefficients of the best matched blocks in the horizontal sub-band of the cover image. Two keys must be communicated: one to hold the indices to the matched blocks in the cover approximation sub-band, and another for the matched blocks in the horizontal sub-band of the cover. Note that the extracted payload is not totally identical to the embedded version as the only embedded and extracted bits belong to the secret image approximation while setting all the data in other sub images to zeros during the reconstruction process.

2.4.4 Adaptive steganography

Adaptive steganography is a special case of the two former methods. It is also known as "Statistics-aware embedding" (Provos & Honeyman, 2003), "Masking" (Johnson & Jajodia, 1998) or "Model-Based" (Sallee, 2003). This method takes statistical global features of the image before attempting to interact with its LSB/DCT coefficients. The statistics will dictate where to make the changes (Kharrazi et al., 2006) and (Tzschoppe et al., 2003). It is characterized by a random adaptive selection of pixels depending on the cover image and the selection of pixels in a block with large local STD, standard deviation. The latter is meant to avoid areas of uniform colour, smooth areas. This behaviour makes adaptive steganography seek images with existing or deliberately added noise and images that demonstrate colour complexity.

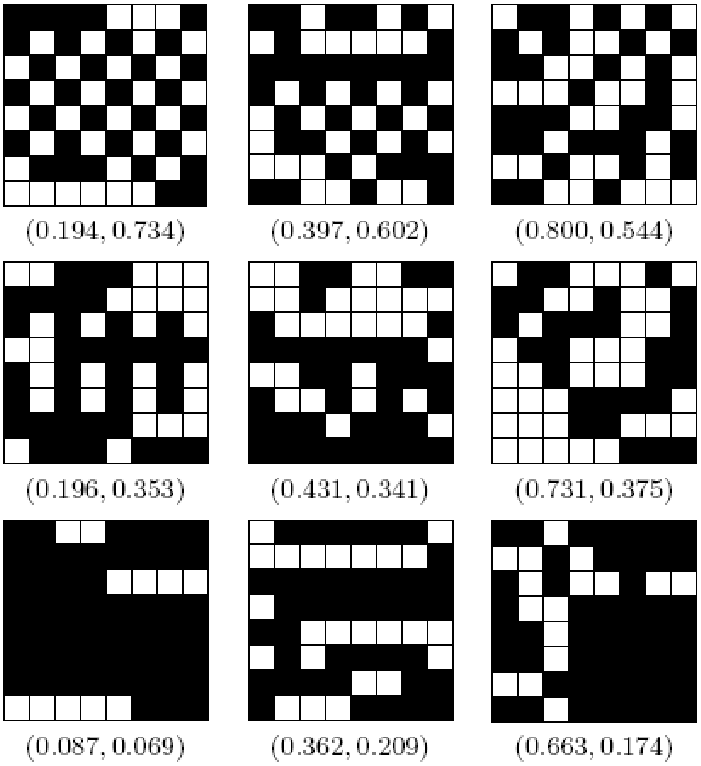
Wayner dedicated a chapter in a book to what he called “life in noise”, pointing to the usefulness of data embedding in noise (Wayner, 2002). It is proven to be robust with respect to compression, cropping and image processing (Fard et al., 2006), (Chang & Tseng, 2004) and (Franz & Schneidewind, 2004). The model-based method, MB1, described in (Sallee, 2003), generates a stego-image based on a given distribution model, using a generalized Cauchy distribution, that results in the minimum distortion.
Due to the lack of a perfect model, this steganographic algorithm can be broken using the first-order statistics (Böhme & Westfeld, 2004) and (Böhme & Westfeld, 2005). Moreover, it can also be detected by the difference of ‘blockiness’ between a stego-image and its estimated image reliably (Yu et al., 2009). The discovery of ‘blockiness’ led the author in (Sallee, 2003) to produce an enhanced version called MB2, a model-based with de-blocking (Sallee, 2005). Unfortunately, even MB2 can be attacked, as highlighted in Section 2.6.

Edge embedding follows edge segment locations of objects in the host gray-scale image in a fixed block fashion each of which has its centre on an edge pixel. Whilst simple, this method is robust to many attacks and it follows that this adaptive method is also an excellent means of hiding data while maintaining a good perceptibility.

Chin-Chen and his colleagues propose an adaptive technique applied to the LSB substitution method. Their idea is to exploit the correlation between neighbouring pixels to estimate the degree of smoothness. They discuss the choices of having 2, 3 and 4 sided matches. The payload, embedding capacity, is high (Chang et al., 2004).

Hioki presented an adaptive method termed “*A Block Complexity based Data Embedding*”, ABCDE (Hioki, 2002). Embedding is performed by replacing selected suitable pixel data of noisy blocks in an image with another noisy block obtained by converting data to be embedded. This suitability is identified by two complexity measures to properly discriminate complex blocks from simple ones, which are run-length irregularity and border noisiness, see Figure 2.19. The left integers in Figure 2.19 denote β for run-length irregularity and the right integers denote γ for border noisiness.

The hidden message is more a part of the image than just added noise (Raja et al., 2008). The ABCDE method introduced a large embedding capacity, however, certain control parameters had to be configured manually, e.g., finding an appropriate section length for sectioning a stream of resource blocks and finding the threshold value that controls identification of complex blocks. These requirements render the method unsuitable for automatic processes.



(0.194, 0.734) (0.397, 0.602) (0.800, 0.544)

(0.196, 0.353) (0.431, 0.341) (0.731, 0.375)

(0.087, 0.069) (0.362, 0.209) (0.663, 0.174)

Figure 2.19: Blocks of various complexity values (Hioki, 2002)

Table 2.2 shows the parameters that the algorithm encompasses. To eliminate fake complex blocks resulting from considering an adjacent Pure Binary Code, PBC, Hioki chose to convert decimals into RBGC. The problem which RBGC was used to solve was the complexity of the higher bit planes to tolerate little relation to the true variation of the image pixels' intensities creating what is often called "hamming cliffs" (Srinivasan, 2003).

Table 2.2: Parameters of ABCDE (Hioki, 2002)

External Parameters
Block size, n x n
External or Internal Parameters
M-sequence parameters
The characteristic polynomial
The initial polynomial
The seed
Threshold values for complexity measures for each bit plane
Internal Parameters
Resource file parameters
The name of the resource file
The size of the resource file
The length of sections

There are two vague issues which are obscurely discussed at the end of Hioki's work. One arises when the carrier image's dimensions are not proportional to the block division scheme and so fragments from these dimensions are kept away from the embedding process. There is no indication by the author of the possible impact of this decision as it might leave a clear contrast between the modified and the intact parts of the image which distorts its statistical properties. The second point is the introduction of the zero padding when the compressed resource file size is not a multiple of the block size. The author did not show any explanation on how to generate complexity from such a compressed file since there will be a sequence of zeros resulting from the "0" padding notion. The author in the experimental section does not show how resilient the algorithm is to different image processing attacks, e.g., rotation, additive noise, cropping, and compression.

Indeed, the ABCDE algorithm provides an improvement over a former method known as BPCS, Bit Plane Complexity Segmentation, (Spaulding et al., 2002), which, in turn, was introduced to compensate for the drawback of the traditional LSB manipulation techniques of data hiding (Fridrich, 1999). The computational complexity of the algorithm to find a phase key that passes the threshold is time consuming and there is no guarantee that it will always evolve into an optimal solution (Srinivasan et al., 2004).

BPCS steganography is not robust to even small changes in the image (Kawaguchi & Eason, 1998), and this weakness is inherited by the ABCDE algorithm also since its underlying framework is based on BPCS. This intolerance to any manipulation of the stego-image is perceived by the authors in (Kawaguchi & Eason, 1998) as a merit. They were over-optimistic about this lack of robustness in the sense that any kind of attack would “destroy the embedded evidence" which points, in their view, to image tampering. Robustness of steganography is one of the three main goals to be achieved and this is definitely not shown in Kawaguchi’s argument. Their algorithm would fail to retrieve the embedded data in two cases: first when the stego-image is attacked resulting in the destruction of the embedded data, and second when an image is plain clear, meaning that no embedding process took place. These two contradictory justifications, due primarily to lack of robustness, would not be appealing characteristics to forensics experts or other interested bodies.

In (Raja et al., 2008), the authors chose to use wavelet transforms that map integers to integers instead of using the conventional Wavelet transforms. This can overcome the difficulty of floating point conversion that occurs after embedding. Their scheme embeds the payload in non overlapping 4x4 blocks of the low frequency, where two pixels at a time are chosen, one on either side of the principal diagonal. Cover image adjustment was required to prevent the problem of under/overflow of pixel values after embedding. In the respective section, they discuss the overflow problem only, where they suggest using the following system prior to embedding:

$$C'(i,j,k) = \begin{cases} C(i,j,k) - (2^N - 1) & \text{if}, C(i,j,k) = 255 \\ C(i,j,k) & \text{Otherwise.} \end{cases} \tag{2.6}$$

where, *C’ (i, j, k)* denotes the modified pixel and *N* represents the number of bits to be embedded in each coefficient, i.e., *N*=4. Hence any value of 255 will be converted to 240. For a true colour image format, they apply the algorithm on each colour plane separately. This step ignores the high correlation between colour planes in natural images. Not taking this phenomenon into consideration means the embedding scenario will corrupt some of the inherited statistics of the cover image, a trap that severely exposes the stego-image to steganalysis attacks. The authors also state some assumptions, embedding is carried out only on non-singular matrices, also ±15 is imperceptible to human vision, finally, the cover image and payload are assumed to be JPEG and the cover be a square matrix of size 512x512. The second assertion is in doubt however. Even though this can be possibly acceptable from a human visual perspective, however, from a statistical point of view, this amount of change is intolerable. Before they conclude, they state that their cover image and stego-image version are similar, even though the best candidate in their experiments has a PSNR, Peak Signal-to-Noise Ratio, that did not exceed 45 dB.

In (Lin et al., 2008) the authors attempt to create a method to restore the marked image to its pristine state after extracting the embedded data. They achieve this by applying the pick-point of a histogram in the difference image to generate an inverse transformation in the spatial domain. The cover image is divided into non-overlapping 4x4 blocks where a difference matrix of size 3x4 is generated for each block. The selection of the local histogram's peak point p_b will direct the embedding process and matrix manipulation. The example shown in their hiding phase section might not be sufficient to verify the accuracy of the algorithm. Some questions remain unanswered such as what happens when there are two peak-points instead of one? On which criterion will the selection be based? Another issue occurs when transforming the matrix SD_b, extracting embedded message in difference image, to RD_b, reconstructed difference image, it is highly likely that after the subtraction process there will be some values that collude with the peak value which confuses the extraction of the embedded data. To prevent over/underflow, caused by the arithmetic operations on values close to boundaries, i.e., [0 255], the authors use the modulus operator, i.e., mod 256. There was no adequate explanation on the effect of homogeneous, dark, bright, and edged blocks on the algorithm efficiency.

In (Wu & Shih, 2006) and (Shih, 2008), a GA (genetic algorithms) based method is presented which generates a stego-image to break the detection of the spatial domain and the frequency-domain steganalysis systems by artificially counterfeiting statistical features. Time complexity, which is usually the drawback of genetic based algorithms, was not discussed. They mentioned that "the process is repeated until a predefined condition is satisfied or a constant number of iterations are reached. The predefined condition is the situation when we can correctly extract the desired hidden message." Again, it was not stated whether the process of determining such a condition was done automatically or involving a human inference, visual perception. The suggested GA-based rounding-error correction algorithm, whilst interesting, still needs proof of generalization. Wu and Shih closed their introduction section by saying, "this is the first paper of utilizing the evolutionary algorithms in the field of steganographic systems" (Wu & Shih, 2006). It should be noted that image hiding using genetic algorithms was known prior to their work such as in (Maity et al., 2004). In (Yu et al., 2009), the authors proposed extending the conventional '±1' algorithm to JPEG images using genetic algorithms.

Kong and his colleagues proposed a content-based image embedding based on segmenting homogenous greyscale areas using a watershed method coupled with Fuzzy C-Means, FCM (Kong et al., 2009). Entropy was then calculated for each region. Entropy values dictated the embedding strength where four LSBs of each of the cover's RGB primaries were used if it exceeded a specific threshold otherwise only two LSBs for each were used. The drawback of this method was its sensitivity to intensity changes which would affect severely the extraction of the correct secret bits. As a side note, in (Kong et al., 2009), the authors also reported the use of a logistic map to encrypt the secret bit stream which seems vulnerable to a Chosen-plaintext attack, CPA.

Chao and his colleagues presented a 3D steganography scheme (Chao et al., 2009). The embedding scheme hides secret messages in the vertices of 3D polygon models. Similarly, Bogomjakov et al. hide a message in the indexed representation of a mesh by permuting the order in which faces and vertices are stored (Bogomjakov et al., 2008). Although, such methods claim higher embedding capacity, time complexity to generate the mesh and rendering can become issues. Moreover 3D graphics are not easily ported compared to digital images.

Nakamura and Zhao, propose a morphing process that takes as input the secret image and the cover file (Nakamura & Zhao, 2008). The method does not discuss the generated features from the cover and secret images used for morphing and how to regenerate them from the stego-image.

Zeki and Azizah proposed what they termed as ‘the intermediate significant bit algorithm’ (Zeki & Manaf, 2009). They studied different ranges of an 8-bit image and found the best compromise for distortion and robustness was in the following range: {0:15} {16:31} … {224:239} {240:255}. The core idea in the embedding process is to find the nearest range that matches the secret bit in the next or previous range.

2.5 Performance Analysis of Methods in the Literature with Recommendations

As a performance measurement for image distortion, the well known Peak-Signal-to-Noise Ratio, PSNR, which is classified under the difference distortion metrics, can be applied to the stego-images. It is defined as:

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{C_{\max}^2}{\mathrm{MSE}}\right) \tag{2.7}$$

where *MSE* denotes the Mean Square Error which is given as:

$$\mathrm{MSE} = \frac{1}{MN}\sum_{x=1}^{M}\sum_{y=1}^{N}\left(S_{xy} - C_{xy}\right)^2 \tag{2.8}$$

where *x* and *y* are the image coordinates, *M* and *N* are the dimensions of the image, S_{xy} is the generated stego-image and C_{xy} is the cover image. Also $C^2{}_{\max}$ holds the maximum value in the image, for example:

$$C_{\max} \le \begin{cases} 1, & \text{double} - \text{precision} \\ 255, & \text{8bit} \end{cases}$$

Many authors such as (Kermani & Jamzad, 2005), (Li & Wang, 2007), (Hashad et al., 2005), (Yu et al., 2007), (Drew & Bergner, 2007), (Saenz et al., 2000) and (Rodriguez & Rowe, 1995), consider $C_{\max}$=255 as a default value for 8-bit images. It can be the case, for instance, that the examined image has only up to 253 or fewer representations of gray colours. Knowing that C_{max} is raised to a power of 2 results in a severe change to the PSNR value. Thus C_{max} can be defined as the actual maximum value rather than the largest possible value. PSNR is often expressed on a logarithmic scale in decibels, dB. PSNR values falling below 30dB indicate a fairly low quality, i.e., distortion caused by embedding can be obvious. A high quality stego-image should strive for a PSNR value of 40dB and above. Table 2.3 shows different PSNR values spawned by various software based on spatial domain methods described in Section 2.5.2 (Online Software, n.d.), applied on the images shown in Figure 2.20, Figure 2.21, Figure 2.22 and Figure 2.23, which depict the output of each of the tools.

Table 2.3: Summary of performance of common software (Kharrazi et al., 2006)

Software	*PSNR*		*Visual Inspection*
	Set A	*Set B*	
[Hide&Seek]	18.608	22.7408	Very clear grainy noise in the stego-image, which renders it the worst performer in this study.
[Hide-in-Picture]	23.866	28.316	Little noise. Accepts only 24-bit bmp files. Creates additional colour palette entries. In this case the original boat image has 32 colours and the generated stego-image augmented the number to 256 by creating new colours.
[Stella]	26.769	16.621	Little noise. Works only with 24-bit images
[S-Tools]	37.775	25.208	No visual evidence of tamper
[Revelation]	23.892	24.381	No visual evidence of tamper, but pair effect appears on the histogram of some outputs

Van Der Weken et al. proposed other Similarity Measures, *SMs* (Van Der Weken et al., 2004). They analysed the efficiency of ten SMs in addition to a modified version of PSNR constructed based on neighbourhood blocks which better adapt to human perception. In order to produce a fair performance comparison between different methods of invisible watermarking, Kutter and Petitcolas discussed a novel measure adapted to the Human Visual System, HVS (Kutter & Petitcolas, 1999). Figure 2.20 shows (Left to right) set A: Cover image Boat, (321x481) and the secret image Tank, (155x151), set B: Cover image Lena (320x480) and secret image Male (77x92), respectively. It is also noted that some algorithms, like the one used in the Revelation software, have the pair effect fingerprint that appears on stego-images.

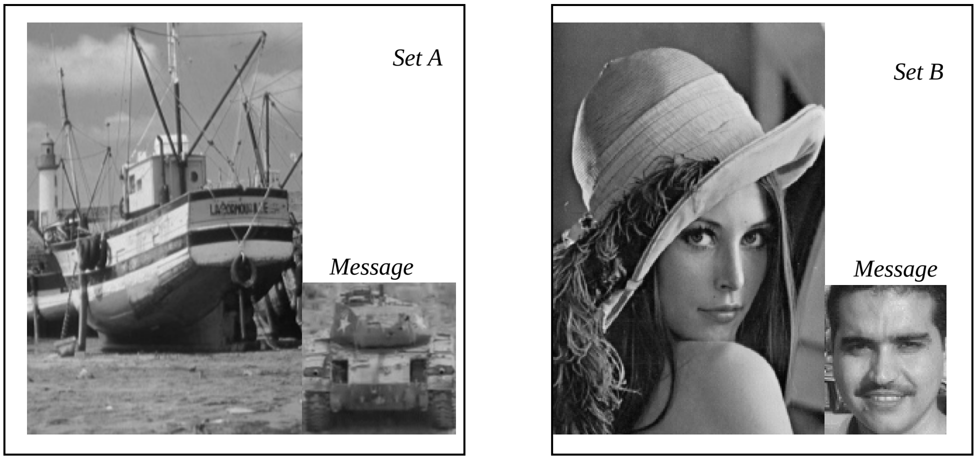


Figure 2.20: Images used to generate Table 2.3

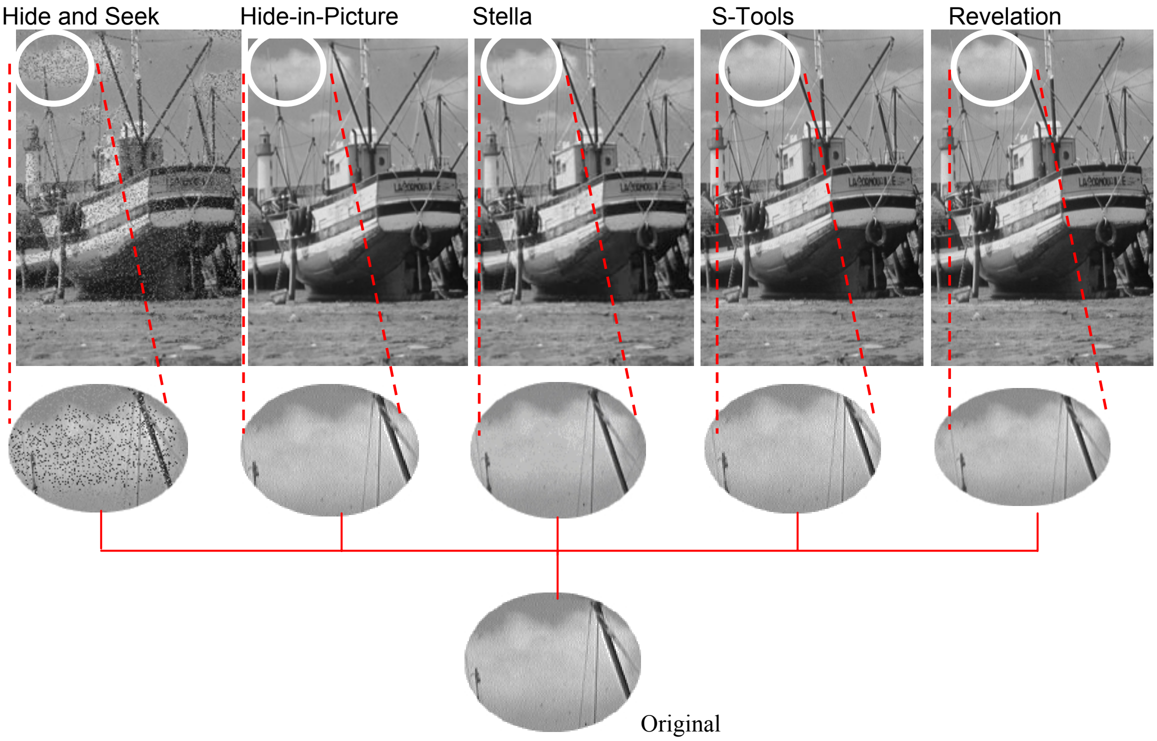


Figure 2.21: Set A: stego-images of each software tool appearing in Table 2.3

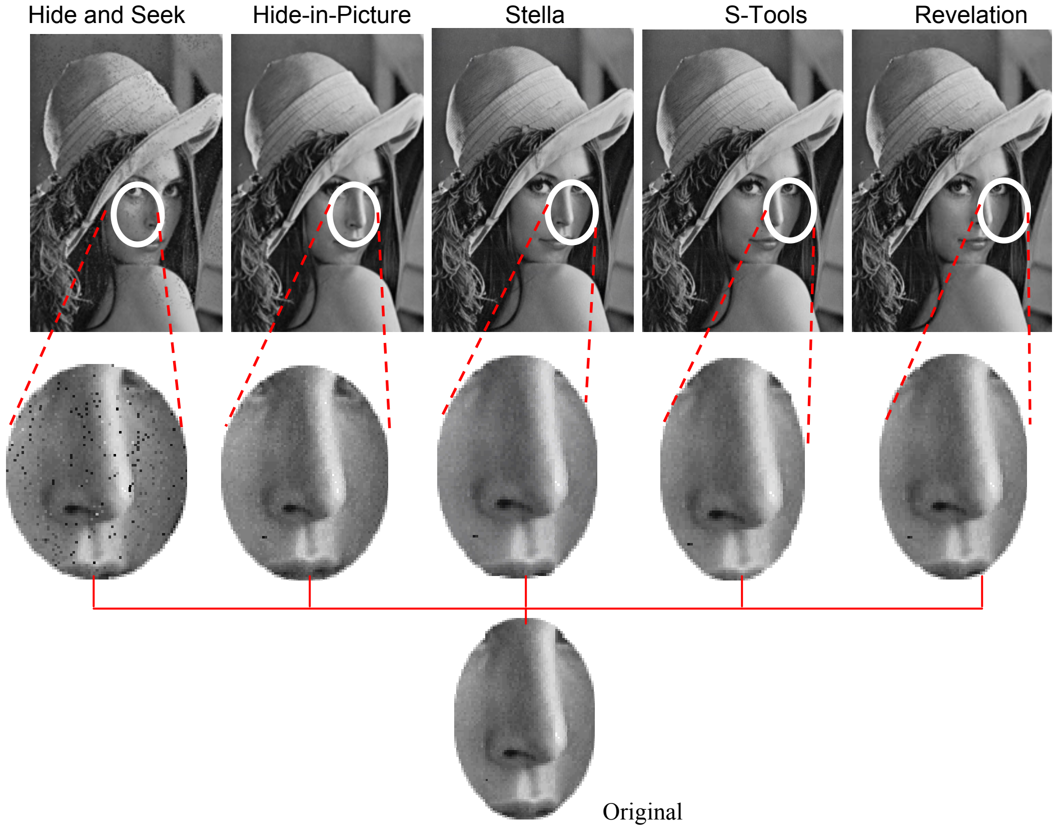


Figure 2.22: Set B: stego-images of each software tool appearing in Table 2.3

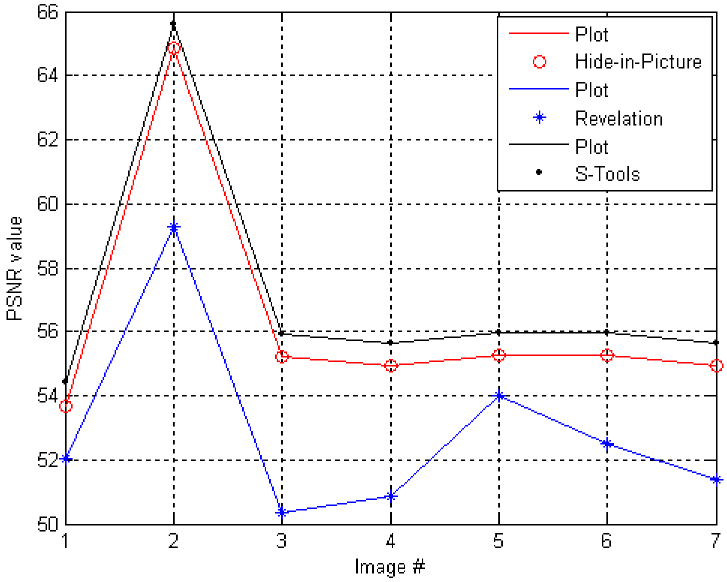


Figure 2.23: Additional experiments on steganography software

Table 2.4 compares some software tools appearing in (Online Software, n.d.) based on:

- the domain on which the algorithm is applied, e.g., spatial or frequency domain,
- the support for encryption,
- random bit selection and
- the different supported image formats.

A performance analysis of some steganographic tools is provided in (Kharrazi et al., 2006). The drawback of the current techniques is also tabulated in Chapter 7, section 7.2. In Table 2.4, the sign (✓) indicates the characteristic is present, (-) denotes unavailability of information, while (x) gives the negative response. In the table columns refer to (1)(2) frequency domain (3) encryption support (4) random bit selection (5) image format. As it is clear from the table, all of the mentioned steganographic algorithms have been detected by steganalysis methods and thus a robust algorithm with a high embedding capacity needs to be investigated.

Table 2.4: Comparison of different tools

Name	*Creator*	*Year*	*(1)*	*(2)*	*(3)*	*(4)*	*(5)*	*Detected by*
JSteg	Derek Upham	-	x	✓ DCT	x	x	JPEG	- X^2-test (Westfeld & Pfitzmann, 1999) - Stegdetect -Fridrich's Algorithm
JSteg-Shell	John Korejwa	-	x	✓ DCT	✓ RC4	-	JPEG	- X^2-test
OutGuess version 0.13b	Provos and Honeyman	-	x	✓ DCT	✓ RC4	✓	JPEG	- X^2-test, extended version - Stegdetect
White Noise Storm	Ray (Arsen) Arachelian	1994	✓	x	✓	✓	PCX	- X^2-test
EZStego	Romana Machado	1996	✓	x	✓	x	BMP, GIF	-RS-steganalysis
S-Tools	Andrew Brown	1996	✓	x	✓ IDEA, DES, 3DES,MPJ2, NSEA	x	BMP, GIF	- X^2-test
JPhide	Allan Latham	1999	x	✓ DCT	✓ Blowfish	x	JPEG	- X^2-test - Stegdetect
OutGuess version 0.2	Provos and Honeyman	2001	x	✓ DCT	✓ RC4	✓	JPEG	-Fridrich's Algorithm
F5	Andreas Westfeld	2001	x	✓	✓	✓	JPEG	-Fridrich's Algorithm

There appears to be two main groups in the area, one for creating steganography algorithms and another group for creating a counter attack, steganalysis. Fard (Fard et al., 2006) state clearly that "there is currently no steganography system which can resist all steganalysis attacks". "Ultimately, image understanding is important for secure adaptive steganography. A human can easily recognize that a pixel is actually a dot above the letter '*i*' and must not be changed. However, it would be very hard to write a computer program capable of making such intelligent decisions in all possible cases, (Fridrich, 1999)". "While there are numerous techniques for embedding large quantities of data in images, there is no known technique for embedding this data in a manner that is robust in light of the variety of manipulations that may occur during image manipulation" (Bender et al., 2000).

"Some researchers proposed to model the cover characteristics and thus create an adaptive steganography algorithm, a goal which is not easily achieved" (Katzenbeisser, 2000). Determining the maximal safe bit-rate that can be embedded in a given image without introducing statistical artefacts remains a very complicated task (Fridrich & Goljan, 2002). The above challenges motivated the steganography community to create a more fundamental approach based on universal properties and adaptive measures (Martin et al., 2005).

To sum up, the following points are noted:
Algorithms F5 and Outguess are the most reliable algorithms although they violate the second order statistics. Both utilise DCT embedding, i.e., they work in the JPEG compression domain, however, unlike first thought, these algorithms are still vulnerable to re-compression.

Embedding in the DWT domain shows promising results and outperforms DCT embedding especially in terms of compression survival (Wayner, 2002). Wavelet smoothing which is a term applied to data filtering in wavelet space, followed by data reconstruction are normal steps in DWT (Murtagh, 2007). Thus a steganographer should be cautious when embedding in the transformation domains in general, however DWT tends to be more flexible than DCT. Unlike JPEG, the introduced image coding system JPEG2000 allows wavelets, i.e., decimated bi-orthogonal wavelet transform, to be employed for compression in lieu of the DCT (JPEG2000, 2007) and (Starck et al., 2007). This makes DWT based steganography the future leading method. An in-depth insight on various wavelet-based applications is provided in (Starck et al., 1998). Without loss of generality, edge embedding maintains an excellent distortion free output whether it is applied in the spatial, DCT or DWT domains (Areepongsa et al., 2000). However, the limited payload is its downfall.

Recognising and tracking elements in a given carrier while embedding can help survive major image processing attacks and compression. This manifests itself as an adaptive intelligent type where the embedding process affects only certain regions of interest, ROI, rather than the entire image. With the boost of Computer Vision, CV, and pattern recognition disciplines this method can be fully automated and unsupervised. These elements, ROIs, e.g., faces in a crowd (Kruus et al., 2003), can be adjusted in perfectly undetectable ways. The majority of steganography research to date has overlooked the fact that utilising objects within images can strengthen the embedding robustness - with few exceptions. A steganography approach reported in (Cheddad et al., 2008e) and (Cheddad et al., 2009c), incorporated computer vision to track and segment skin regions for embedding under the assumption that skin tone colour provides better embedding imperceptibility. They used computer vision techniques to introduce their rotation and translation invariance embedding scheme to establish an object oriented embedding, OOE. A related method, in the sense that it uses objects in images, meant for watermarking, was introduced by authors in (Nikolaidis & Pitas, 2001) and (Nikolaidis & Pitas, 2000). In this method they employed an adaptive clustering technique which derived a robust region representation of the original image. The robust regions were approximated by ellipsoids, whose bounding rectangles were chosen as the embedding area for the watermark.

Most of the existing steganographic methods rely on two factors: the secret key and the robustness of the steganographic algorithm. However, all of them either do not address the issue of encryption of the payload prior to embedding or merely give a hint of using one or more of the conventional block cipher algorithms. Hence, Westfeld et al. concluded their CRYSTAL project with an important observation that “Crypto-Stego interaction is not very well researched, yet” (CRYSTAL, 2004). Authors of (Shih, 2008),(Lou & Sung, 2004) and (Cheddad et al., 2008d) are among the few who discuss in detail the encryption of the payload prior to embedding.

There are some basic points that should be noted by a steganographer:
In order to eliminate the attack of comparing the original image file with the stego-image, where a very simple kind of steganalysis is essential, we can freshly create an image and destroy it after generating the stego-image. Embedding into images available on the World Wide Web is not advisable as a steganalysis devotee might notice and opportunistically utilize them to decode the stego-image.

In order to avoid any Human Visual Perceptual attack, the generated stego-image must not have visual artefacts. Alteration made up to the 4th LSB of a given pixel will yield a dramatic change in its value. Such an unwise choice on the part of the steganographer will thwart the perceptual security of the transmission. Consider the following example: let a pixel intensity value be 173, which in binary is $(1010\mathbf{1}101)_2$. If the secret bit is ‘0’ then the stego-image pixel will be 165, $(1010\mathbf{0}101)_2$ in binary, or 172, $(10101100)_2$ in binary.

Smooth homogeneous areas must be avoided, e.g., cloudless blue sky over a blanket of snow, however chaotic areas with naturally redundant noisy backgrounds and salient rigid edges should be targeted (Johnson & Katzenbeisser, 2000) and (Wu & Tsai, 2003). This point, however, needs further investigation as some authors think differently. An example is the study of Kodovsky and Fridrich that concludes “texture-adaptive selection channels do not improve steganographic security” (Kodovsky & Fridrich, 2008a).

The secret data must be a composite of balanced bit values (Socek et al., 2007), since in general, the expected probabilities of bit 0 and bit 1 for a typical cover image are the same, i.e., $P\{0\} = P\{1\} = 0.5$ (Chen & Wang, 2009). In some cases, encryption provides such a balance.

It is essential that encryption not only is able to offer such a balance but also is random enough so that it can mimic the LSBs of the cover image. Even though Wayner has answered the question "how random is the noise?" qualitatively, (Wayner, 2002, p.26), there are various methods which estimate randomness quantitatively, see, (Rukhin et al., 2008). One way to measure such randomness is to use the Cross-Covariance as illustrated in Chapter 6.

The last LSB where the stego-value, compared to the plain-value, is unchanged, increased or decreased by one, change by ± 1 in the 1st LSB or ± 4 in the 3rd LSB, eventually leaves traceable statistical violations. Many algorithms to date still use such conventional models either in the spatial domain or the transform domain. The RBGC allows alteration to even the third LSB, i.e., change by ± 3, in the DWT without much degradation compared to the conventional use of PBC.

2.6 Steganalysis

This section presents a brief description and some standards that a steganographer should usually examine. Steganalysis is the science of attacking steganography in a battle that never ends. It mimics the already established science of Cryptanalysis. Note that steganographers can create a steganalysis system merely to test the strength of their algorithm. Steganalysis is achieved through applying different image processing techniques, e.g., image filtering, rotation, cropping, and translation. More deliberately, it can be achieved by coding a program that examines the stego-image structure and measures its statistical properties, e.g., first order statistics, histograms, or second order statistics, correlations between pixels, distance and direction. JPEG double compression and the distribution of DCT coefficients can give hints on the use of DCT-based image steganography. Passive steganalysis attempts to destroy any trace of secret communication, without detecting the secret data, by using the above mentioned image processing techniques: changing the image format, flipping all LSBs or by undertaking a severe lossy compression, e.g., JPEG. Active steganalysis however, is any specialized algorithm that detects the existence of stego-images.

Spatial steganography generates unusual patterns such as sorting of colour palettes, relationships between indexed colours and exaggerated "noise", as can be seen in Figure 2.24, all of which leave traces to be picked up by steganalysis tools. This method is very fragile (Marvel & Retter, 1998). The figure shows: (left to right) original image, LSBs of the image before embedding and after embedding, respectively (Bas, 2003, pp.16-17).

> *LSB encoding is extremely sensitive to any kind of filtering or manipulation of the stego-image. Scaling, rotation, cropping, addition of noise, or lossy compression to the stego-image is very likely to destroy the message. Furthermore an attacker can easily remove the message by removing, zeroing, the entire LSB plane with very little change in the perceptual quality of the modified stego-image* (Lin & Delp, 1999).

Almost any filtering process will alter the values of many of the LSBs (Anderson & Petitcolas, 1998).

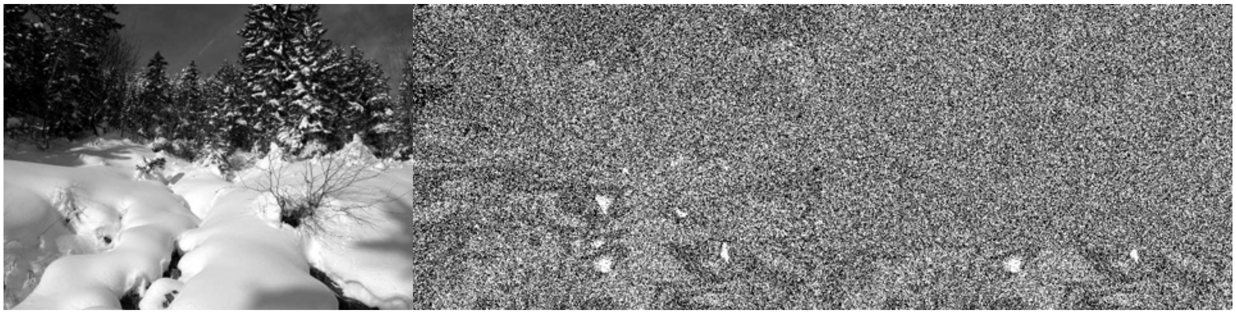


Figure 2.24: Steganalysis using visual inspection (Lin & Delp, 1999)

By inspecting the inner structure of the LSBs, Fridrich and her colleagues claimed to be able to extract hidden messages as short as 0.03bpp, bit per pixel (Fridrich et al., 2001b). Kong et al. stated that the LSB methods can result in the "pair effect" in the image histograms (Kong et al., 2005). As can be seen in Figure 2.25, this "pair effect" phenomenon is empirically observed in steganography based on the modulus operator.

Figure 2.25 shows: (top) original and (bottom) stego-image. Note that it is not always the case that modulus steganography produces such a noticeable phenomenon. This operator acts as a means to generate random locations, i.e. not sequential, to embed data. It can be a complicated process or a simple one like testing, in a raster scan fashion, if a pixel value is even then embed, otherwise do nothing. Avcibas et al. applied binary similarity measures and multivariate regression to detect what they call "telltale marks" generated by the 7th and 8th bit planes of a stego-image (Avcibas et al., 2002).

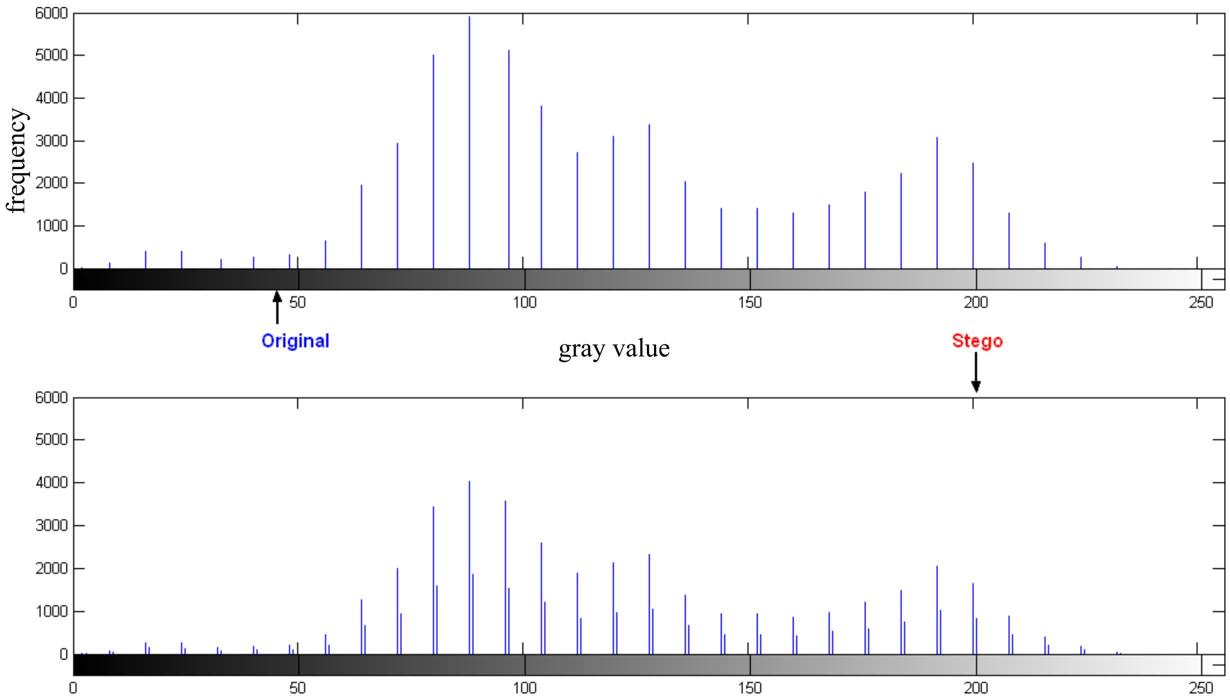
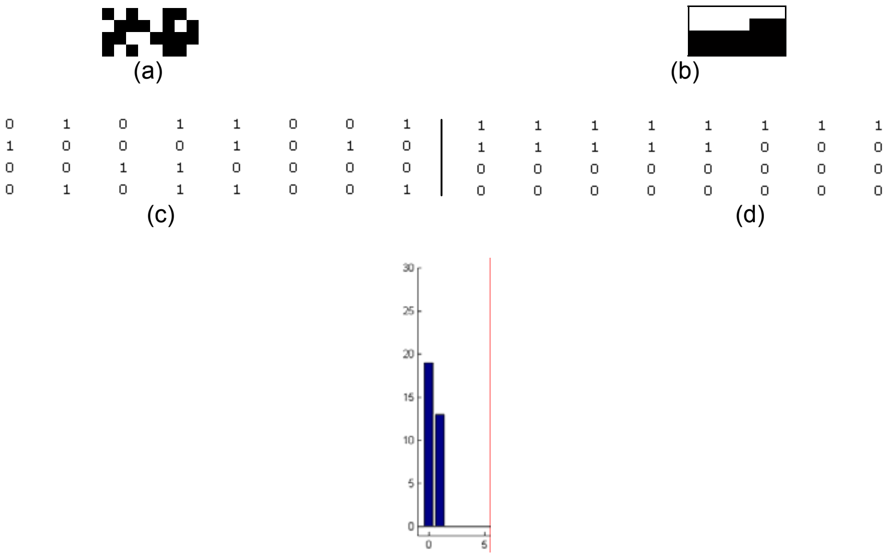


Figure 2.25: Histograms demonstrating the "*pair effect*"

The histograms in Figure 2.25 are given by the following discrete function:

$$H(k_i) = \sum_{i=0}^{255} g(k_i) \tag{2.9}$$

where, k_i is the i^{th} intensity level in the interval {0, 255} and $g(k_i)$ is the number of pixels in the image whose intensity level is k_i . It is the nature of standard intensity image histograms to track and graph frequencies of pixel values in a given image and not their structure and how they are arranged, see Figure 2.26.



(a) (b)

0	1	0	1	1	0	0	1
1	0	0	0	1	0	1	0
0	0	1	1	0	0	0	0
0	1	0	1	1	0	0	1

(c)

1	1	1	1	1	1	1	1
1	1	1	1	1	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

(d)

(e)

Figure 2.26: Standard histograms may not reveal the structure of data. The figure depicts: *(a)* an 8x4 matrix stored in double precision and viewed *(b)* another transformed image of (a) with the same histogram *(c)* pixel values of (a) *(d)* pixel values of (b) and *(e)* the histogram which describes both matrices

Chi-square, χ^2, and Pair-analysis algorithms can easily attack methods based on the spatial domain. The Chi-square algorithm is non-parametric, a rough estimate of confidence, statistical algorithm used to detect whether the intensity levels scatter in a uniform distribution throughout the image surface or not (Civicioglu et al., 2004). If one intensity level has been detected as such, then the pixels associated with this intensity level are considered as corrupted pixels or in this case have a higher probability of having embedded data. The classical Chi-square algorithm can be fooled by randomly embedded messages, thus Bohne and Westfeld developed a steganalysis method to detect randomly scattered hidden data in the LSB spatial domain that applies the Preserving Statistical Properties, PSP, algorithm (Böhme & Westfeld, 2005).

If $o_i = \{o_1, o_2, ..., o_n\}$ denotes the observed data which can be seen as the number of times the symbols 1, 0 occur in the image LSBs (Wayner, 2002, p.311), let e_i denote the number of times the event is expected to occur. Therefore, the test statistic is of the form:

$$\chi^2 = \sum \frac{(o_i - e_i)^2}{e_i} \tag{2.10}$$

To avoid detection during steganalysis attacks, Fu and Au (Fu & Au, 2002) and Guo (Guo, 2008), in watermarking, proposed data hiding methods for halftone images. The assumption here is that the inverse halftoning process would smooth the noise occurring from data embedding. However, inspired by the steganalysis techniques for gray level images, Cheng and Kot successfully created a system able to counter-attack such methods by exploiting the wavelet statistic features extracted from the reconstructed gray level images through the inverse halftoning of a given halftone image fed into the Support Vector Machine's classifier (Cheng & Kot, 2009).

Fridrich et al. propose a statistical method that uses higher-order statistics called RS steganalysis, also called *dual statistics* (Fridrich et al., 2001a). These statistics provide an estimated percentage of flipped pixels caused by embedding as can be seen from Table 2.5 generated from Figure 2.27. Here, image blocks, usually a 2x2, are classified as *regular*, *R*, or *singular*, *S,* depending on the increase or decrease of noise within each block, respectively. This classification is repeated using the dual form of embedding, $1 \leftrightarrow 2, 3 \leftrightarrow 4, ..., 255 \leftrightarrow 0$, and *R'* and *S'* are generated. In natural clean images the following assumptions hold: *R'= R and S'= S* (Fridrich et al., 2001a).

Table 2.5: RS estimations, table shows the estimated number of pixels with flipped LSBs for the test image, with the actual numbers that should be detected in an ideal case, indicated in parentheses (Fridrich et al., 2001a)

Image	*Red (%)*	*Green (%)*	*Blue (%)*
Cover image	2.5 (0.0)	2.4 (0.0)	2.6 (0.0)
Steganos	10.6 (9.8)	13.3 (9.9)	12.4 (9.8)
S-Tools	13.4 (10.2)	11.4 (10.2)	10.3 (10.2)
Hide4PGP	12.9 (10.0)	13.8 (10.1)	13.0 (10.0)

Figure 2.27: A test image for the RS steganalysis' performance (Fridrich et al., 2001a)

Cancelli et al. reveal that the performance of current state-of-the-art steganalysis algorithms for detection of ± 1 steganography is highly sensitive to the used training and testing databases (Cancelli et al., 2008). Their experiments also show that the examined algorithms are not applicable in their current state since the embedding rate for testing is very likely to be unknown, while it was assumed otherwise in those algorithms. Therefore, they conclude that no single steganalysis algorithm is constantly superior.

In the frequency domain, Pevny and Fridrich developed a multi-class JPEG steganalysis system that comprised DCT features and calibrated[3] Markov features, which were then merged to produce a 274-dimensional feature vector (Pevny & Fridrich, 2007). This vector is fed into a Support Vector Machine multi-classifier capable of detecting the presence of Model-Based steganography, F5, OutGuess, Steghide and JP Hide&Seek. Li et al. exposed some of the weaknesses in the '*YASS*', *Yet Another Steganograhic System*, proposed in (Solanki et al., 2007), by noticing that it introduces extra zero coefficients into the embedded host blocks because of the use of a Quantization Index Modulation, QIM, method and by contrasting statistical features derived from different blocks in the stego-image (Li et al., 2008a).

Targeted embedding methods, such as the new enhanced MB2, are faced with much more accurate targeted attacks. That is because "if the selection channel is public, the

---

[3] Calibration works by subtracting a reference image (obtained by decompressing, cropping and recompressing the stego-image) from the original stego-image (Kodovský & Fridrich, 2009) .

attacker can focus on areas that were likely modified and use those less likely to have been modified for comparison/calibration purposes” (Kodovsky & Fridrich, 2008a, p.6). In (Ullerich & Westfeld, 2007), the authors successfully attacked MB2 using coefficient types that are derived from the blockiness adjustment of MB2. They adapt Sallee's Cauchy model itself to detect Cauchy model-based embedded messages. In (Chen & Shi, 2008), the authors attacked MB2 and other JPEG-based algorithms using Markov process, MP, that exploits the intra-block and inter-block correlations among JPEG coefficients. Vulnerability of pixel-value differencing methods was revealed through histogram analysis (Zhang & Wang, 2004).

2.7 Summary

This chapter presented a background discussion on the major algorithms of steganography deployed in digital imaging. The emerging techniques such as DCT, DWT and Adaptive steganography are not too prone to attacks, especially when the hidden message is small. This is because they alter coefficients in the transform domain, thus image distortion is kept to a minimum. Generally these methods tend to have a lower payload compared to spatial domain algorithms. There are different ways to reduce the bits needed to encode a hidden message. Apparent methods can be compression or correlated steganography, as proposed in (Zheng & Cox, 2007), which is based on the conditional entropy of the message given the cover. In short, there has always been a trade-off between robustness and payload.

Scholars differ about the importance of robustness in steganography system design. Cox regards steganography as a process that should not consider robustness as it is then difficult to differentiate from watermarking (Cox, 2009). Katzenbeisser, on the other hand, dedicated a sub-section to robust steganography. He mentioned that robustness is a practical requirement for a steganography system. “Many steganography systems are designed to be robust against a specific class of mapping.” (Katzenbeisser, 2000, p.32). It is also rational to create an undetectable steganography algorithm that is capable of resisting common image processing manipulations that might occur by accident and not necessarily via an attack. Cox's view is formed based on his definition of steganography and its scope, while Katzenbeisser is looking at the process of steganography in a different way, preferring to view it as a robust secret communication mechanism. This Chapter offered some guidelines and recommendations on the design of a steganographic system.

CHAPTER

THREE

Image Encryption Methods and Skin Tone Detection Algorithms

Most of the existing steganographic methods rely on two factors: the secret key and the embedding strategy. However, all of them either do not address the issue of encryption of the payload prior to embedding or merely use one or more of the conventional block cipher algorithms. Hence, Westfeld and his colleagues concluded their CRYSTAL, CRYptography and encoding in the context of STeganographic Algorithms, project with an important observation that “Crypto-Stego interaction is not very well researched yet” (CRYSTAL, 2004).

Since this thesis advocates an object oriented embedding approach to steganography, which provides an automatic solution to various problems, skin-tone detection is used due to some basic advantages that it provides. These include invariance to rotation and translation, stable middle range chrominance values and fast automatic extraction of non-smooth areas. These advantages will be specifically discussed in Chapter 4.

This Chapter reviews different methods available in the literature for both image encryption and skin-tone segmentation.

3.1 Image Encryption Methods

Unlike text encryption, image encryption is relatively new and has received considerable attention from researchers in recent years who work on secure video and image transmissions. Three types of image encryption are discussed in this Section, block ciphers, chaotic-based ciphers and stream ciphers.

The renowned generic block cipher algorithms, such as Data Encryption Standard, DES, Advanced Encryption Standard, AES and International Data Encryption Algorithm, IDEA, are not suitable for handling bulky data like that of digital images, due to their intensive computational process (Usman et al., 2007) and (Zeghid et al., 2006) unless accelerated by hardware implementations. Additionally, such symmetric-key cryptographic algorithms are found unfit for digital images characterized with some intrinsic features such as bulk data capacity and high pixel correlation and redundancy, (Patidar et al., 2009) and (Chen & Zheng, 2005), especially when confidentiality is required. Other limitations were reported in (Mao & Wu, 2006) in light of multimedia communication, delegate service scenario, such as rate adaptation for multimedia transmission in heterogeneous networks and DC-image extraction for multimedia content searching which cannot be applied directly in the bit-stream encrypted by these cryptographic algorithms. In the transmission and decoding process, standard encryption schemes prove to be an overhead (Yekkala et al., 2007).

Security systems are built on increasingly strong cryptographic methods that foil pattern and statistical analysis attempts (SHA, 2001). Encryption is particularly useful for Intellectual Property Management and Protection, IPMP, standardization group and multimedia communications that prefer handling media streams compliant to certain multimedia coding standards, such as the lossy compressed image type format JPEG, Joint Photographic Experts Group, or the different versions of the Moving Picture Experts Group, MPEG-1/2/4, standard (Wen et al., 2002).

The research in the literature on the design of secure image encryption tends to focus on transferring images into chaotic maps. Chaos theory, which essentially emerged from mathematics and physics, deals with the behaviour of certain nonlinear dynamic systems that exhibit a phenomenon under certain conditions known as chaos which adopt the Shannon requirement on diffusion and confusion (Shih, 2008). Confusion is a property that refers to making the relationship between the description of the key and the statistics of the cipher complex and is achieved through rearranging pixels so that redundancy in the plain-image is spread out over the complete cipher. Diffusion refers to the property that the redundancy in the statistics of the plain-image is "dissipated" in the statistics of the cipher (Shannon, 1949). Due to their attractive features such as sensitivity to initial conditions and random-like outspreading behaviour, chaotic maps are employed for various applications of data protection (Wang et al., 2008). In the realm of 2D data, Shih outlines the following method, called the *Toral automorphism* map, in order to spread the neighbouring pixels into largely dispersed locations (Shih, 2008). The transformation is represented through the following formula:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ l & l+1 \end{bmatrix} * \begin{bmatrix} x \\ y \end{bmatrix} \mod N \tag{3.1}$$

where, $\det\left(\begin{bmatrix} 1 & 1 \\ l & l+1 \end{bmatrix}\right) = 1$ or -1, and *l* and *N* denote an arbitrary integer and the width of a square image respectively. Also, *x* and *y* represent the original pixel coordinates and *x'* and *y'* the new location coordinates. The determinant is referred to as 'det'. Figure 3.1 shows an example of chaotic map. Applying Equation (3.1) to the sample image 'Lena', it can be seen that after exactly 17 iterations, termed as the stable orbit, the chaotic map converged into the original image.

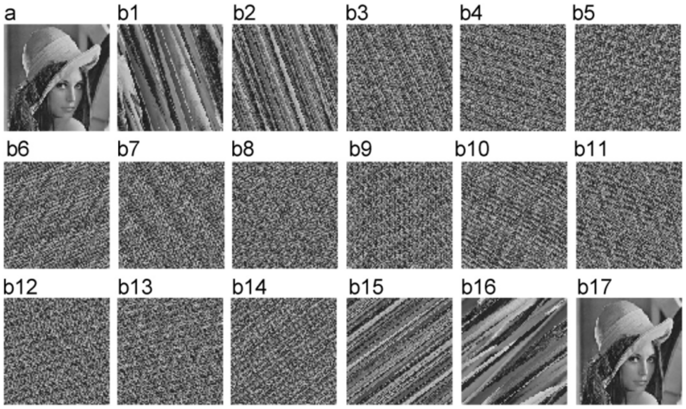


Figure 3.1: An example of chaotic map. (a) the original image, and (b1)–(b17) the relocated images (Wu & Shih, 2006)

This Discrete Time Dynamic System, DTDS, is also the basic framework used in (Lou & Sung, 2004). Regarding this method, it is important to note that since the algorithm uses a determinant in its process, the input matrix can only be square. This constraint was highlighted also in (Usman et al., 2007). A work around this problem might be to apply the algorithm on square blocks of a given image repetitively. However, that would generate noticeable peculiar periodic square patterns given the nature of the process and of course this is not an interesting fact as it conflicts with the aim of generating chaotic maps.

As far as security systems are concerned, the convergence of the translated pixels into their initial locations, i.e., image exact reconstruction after some iterations, is also not an appealing factor. This is an observed phenomenon in a variety of chaotic based algorithms. Given one of the iterations is used, if an attacker gains knowledge of the algorithm and obtains the parameter "*l*", which is actually not difficult to crack using a brute force attack, the attacker will be able to add further iterations which will reveal the original image. For example, Wang et al. (Wang et al., 2007) show that for such systems if two parameters are set to 10 and 8, then regardless of image contents, any image with the dimensions of 256 x 256 will converge after 128 iterations. This periodicity brings insecurity to the process as methods for computing the periodicity can be formulated such as that proposed in (Ashtiyani et al., 2008) and (Bing & Jia-wei, 2005).

In a more detailed and concise attempt to introduce image encryption, Pisarchik (Pisarchik et al., 2006) demonstrated that any image can be represented as a lattice of pixels, each of which has a particular colour in the RGB colour space. The pixel colour is the combination of three components: red, green, and blue, each of which takes an integer value C= (C_r, C_g, and C_b) between 0 and 255. Thus, they create three parallel CMLs, Chaotic Map lattices, by converting each of these three colour components to the corresponding values of the map variable, $x_c = (x_c^r, x_c^g, x_c^b)$ and use these values as the initial conditions, $x_c = x_0$. Starting from different initial conditions, each chaotic map in the CMLs, after a small number of iterations, yields a different value from the initial conditions, and hence the image becomes indistinguishable because of an exponential divergence of chaotic trajectories (Pisarchik et al., 2006). They introduced seven steps for encrypting images and seven steps for decryption. Moreover, four parameters were used of which two were regulated. Their settings can have a tremendous effect on the chaotic map quality. Therefore, the receiver must know the decryption algorithm and the parameters which act as secret keys.

The algorithm is well formulated and adequately presented, it yields good results for RGB images as proclaimed by the authors. It was noticed that they used a rounding operator which was applied recursively along the different iterations. The major concern would be in recovering the exact intensity values of the input image as the recovered image shown in their work might be just an approximation because of the aforementioned operator. This is important, especially in the application of steganography where the objective is to recover the exact embedded file rather than its approximation. The raised point was remarked independently in (Kanso & Smaoui, 2009) where they stated that a sensitive generator, i.e., a generator with a rounding operator, can produce two different binary sequences, after some iterations, for the same initial values and parameters if generated on two different machines which round off fractions after unmatched decimal places. However, a desired algorithm must be efficient, repeatable and portable, that is it works in the same way in different software and hardware environments, (L'Ecuyer, 2006). As a result, such a chaotic encryption system is not invertible under double precision arithmetic (Solak & Çokal, 2008).

Usman (Usman et al., 2007) describe a method for generating chaotic maps to encrypt medical images by repetitive pixel arrangement and column and row permutations. The pixel arrangement is achieved through the following system:

$$\begin{aligned} & \mathrm{X}(i,j) \rightarrow \mathrm{Y}(k,l), \text{where} \\ & k = \lfloor (j+(i-1)N-1)/L \rfloor + 1 \\ & l = (j+(i-1)N-1)\bmod(L)+1 \end{aligned} \tag{3.2}$$

Here, *k*, *l* denote the mapped spatial coordinates of the original location at *i*, *j*. *N* and *L* are the height of the original image and transformed image respectively in such a way that: $(KxL) = (MxN), \text{where}: K \neq M$. The authors show some experiments in which the deciphered phase was missing. It is suspected that the rounding operator introduced in Equation (3.2) will force some pixels to collude at the same location resulting in the loss of information needed for the original image reconstruction. Zou (Zou et al., 2005) reduce the number of iterations using 2D generalised Baker transformations to enhance the key space.

Ultimately the aforementioned methods scramble image pixels using some control parameters and a number of iterations. It is worth noting here that there are several similar image encryption methods using chaotic maps introduced in the literature. The most popular ones are Arnold Cat Map, Baker Map and Tent Map. For in-depth discussions on these maps the reader is referred to the work in (Fridrich, 1997). Fridrich (Fridrich, 1997) uses a block cipher with a private key. The encryption is initialised with a 2D chaotic map which is discretised so that it maps a rectangular lattice of pixels in a bijective manner. Finally, the map is extended to three dimensions to modify the gray levels.

A survey on image encryption is provided in (Shujun et al., 2004). Their review starts with a brief on the need for image and video encryption followed by image encryption techniques. They concluded their work with eight remarks which are listed below:

- permutation-only image and video encryption schemes are generally insecure against known and chosen-plaintext attacks.
- secret permutation is not a prerequisite
- cipher-text feedback is very useful for enhancing the security
- cipher-text feedback can be enhanced further if combined with permutation
- combining a simple stream cipher and a simple block cipher can help improve security
- the diffusion methods used in most chaos-based encryption schemes are too slow
- selective encryption may provide enough security given the dependencies between the unencrypted and encrypted data

- a recommendation to use a slow, but stronger, cipher to encrypt selective data and fast, but weaker, cipher to encrypt the remaining data

Generally speaking chaos algorithms keep image statistics intact and as a result pixels' intensities remain the same. However, the close relationship between chaos and cryptography makes chaos-based cryptographic algorithms a natural candidate for secure communication (Ashtiyani et al., 2008). Shannon's two requirements, confusion and diffusion, must be met when attempting to create any secure cipher algorithm (Shannon, 1949). The Arnold Cat Map, given its nature of data scrambling, satisfies the first requirement but not the second as it was stated earlier that pixel values are not changed. Unfortunately, most chaotic maps are unstable due to the periodicity of the mapping (Lou & Sung, 2004) and (Huang & Feng, 2009). Systems based on these maps are prone to attacks, such as the broken system shown in (Cokal & Solak, 2009).

Other types of image encryption include the Fourier plane encoding algorithm, introduced in (Refregier & Javidi, 1995), which encrypted an image by using two statistically independent random phase codes in the input plane and Fourier plane. The image is multiplied by the first generated code, and then the product is Fourier transformed and multiplied by the second random phase code. This algorithm is attacked in (Gopinathan et al., 2005) using an initial guess of the Fourier plane random phase while searching over a key space to minimise a cost function between the decrypted image for a given key and the original image. This spurred a variety of authors to apply the Fourier transform such as those of (Singh et al., 2008) and(Joshi et al., 2008).

Shin and Kim (Shin & Kim, 2006) presented a phase-only encryption scheme using the Fourier plane. To generate this phase encrypted data, a zero-padded original image, multiplied by a random phase image, was Fourier transformed and its real-valued data is encrypted with key data by using phase-encoded XOR rules. Since the original information is encrypted on the Fourier plane, the decryption cannot retrieve the original image without perceptual degradation, i.e., the PSNR is in the interval [20dB 42.23dB].

One time pad hash algorithms, known also as stream ciphers, were believed to be unsuitable for image encryption since they would require a key of the size of the ciphered image itself (Usman et al., 2007). Sinha and Singh (Sinha & Singh, 2003) used MD5, Message Digest 5, to generate image signatures by which they encrypted the image itself using a bitwise exclusive-OR, XOR, operation. They coupled that with an error control code, i.e., Bose-Chaudhuri Hochquenghem, BCH. The ciphered image was larger than the original because of the added redundancy due to applying the BCH. Since the message digest was smaller than the image, they XOR the signature block by block which eventually left some traces of repetitive patterns. Hence, their method was commented on in (Encinas & Dominguez, 2006) in which they showed also how insecure the method was in some experiments, a fact that provoked Sinha and Singh (Sinha & Singh, 2003) to debate the arguments in their recent published reply in (Sinha & Singh, 2006).

Martinian (Martinian et al., 2005) derived an encryption key from a user's biometric image itself. The reported advantage was that unlike normal passwords, the key was never clearly stored and the user would not need to remember it. However, on one hand, this scheme has a potential flaw if the biometric image is stolen which unlike passwords is impossible to replace. On the other hand the same biometric can be grabbed with different intensities depending on intrinsic factors such as camera model, resolutions, or extrinsic aspects such as environment changes, light, which will lead the encryption algorithm to behave differently.

Gao and Chen (Gao & Chen, 2008) propose an image encryption algorithm based on hyper-chaos, which uses a matrix permutation to shuffle the pixel positions of the plain-image, a logistic map, and then the states combination of hyper-chaos is used to change the grey values of the shuffled-image, diffusion. Their proposed algorithm did not survive attacks for too long. Rhouma and Belghith (Rhouma & Belghith, 2008) successfully broke their cryptosystem using a chosen plaintext attack and a chosen cipher-text attack that recovered the ciphered-image without any knowledge of the key value.

Zeghid (Zeghid et al., 2006) propose a new modified version of AES which involves the design of a secure symmetric image encryption technique. The AES is extended to support a key stream generator for image encryption which can overcome the problem of textured zones existing in other known encryption algorithms. The problems of AES based algorithms are: computational time complexity, generation of repetitive spatial patterns, and the sensitivity to image manipulation.

In the realm of information hiding some steganographic applications prefer to use conventional pseudo-random number generator, PRNG, algorithms which form the basic and essential ingredient for any stochastic simulation in which random variables and other random objects are simulated by deterministic algorithms (L'Ecuyer, 2006).

3.2 Skin Tone Detection Methods

Detecting human skin tone is of utmost importance in numerous applications such as, video surveillance, face and gesture recognition, human computer interaction, human pose modelling, image and video indexing and retrieval, image editing, vehicle drivers' drowsiness detection, controlling users' browsing behaviour, e.g., surfing indecent sites, and steganography. It is regarded as a two-class classification problem, and has received considerable attention from researchers in recent years (Corey et al., 2007) and (Khan et al., 2002), especially those working in the area of biometrics or computer vision.

According to Zhao (Zhao et al., 2007), there are two critical issues for colour-based skin detection: (1) what colour space should be selected? and (2) what segmentation method should be used? This review and the proposed enhancement in Chapter 4 tackle the former issue.

Colour transformations are of paramount importance in computer vision. There exist several colour spaces including: RGB, HSV (Hue, Saturation and Value), HIS (Hue Intensity and Saturation), YIQ (luminance (*Y*) and chrominance (*I* and *Q*)), YC_bC_r

(luminance (Y) and chrominance (C_b and C_r)) (Gomez, 2002). The native representation of colour images is the RGB colour space which describes the world view in three colour matrices: Red (R), Green (G) and Blue (B), see the website dedicated to colour analysis with tools for multi-spectral image analysis (Couleur, 2008). Luminance is present in this space and thus various transforms are used to extract it.

Modelling skin colour implies the identification of a suitable colour space and the careful setting of rules for cropping clusters associated with skin colour. The attempt to attribute numbers to the brain's reaction to visual stimuli is very difficult, hence the aim of colour spaces attempt to describe colour, either between people or between machines or programs (Ford & Roberts, 1998).

Unfortunately, most approaches to date tend to put the illumination channel in the "non useful" zone and therefore act instead on colour transformation spaces that de-correlate luminance and chrominance components from an RGB image. It is important to note that illumination and luminance are defined slightly differently as they depend on each other. As this may cause confusion, for simplicity, these are both referred to here as the function of response to incident light flux or the brightness.

Abadpour and Kasaei (Abadpour & Kasaei, 2005) concluded that "in the YUV, YIQ, and YC_bC_r colour spaces, removing the illumination related component (Y) increases the performance of skin detection process". Others were in favour of dropping luminance prior to any processing as they were convinced that the mixing of chrominance and luminance data makes RGB basis marred and not a very favourable choice for colour analysis and colour based recognition (Hsu et al., 2002) and (Vezhnevets et al., 2003). Therefore, luminance and chrominance are always difficult to tease apart unless the RGB components are transformed into other colour spaces, and even then these spaces do not guarantee total control over luminance. Comprehensive work exists which discusses in depth the different colour spaces and their associated performance (Abadpour & Kasaei, 2005), (Martinkauppi et al., 2001) and (Phung et al., 2005).

Albiol (Albiol et al., 2001) and Hsieh (Hsieh et al., 2002) show that choosing a colour space has no implication on the detection of skin tone given that an optimum skin detector is used. In other words all colour spaces perform similarly. Analogous to this, Phung (Phung et al., 2005) show that skin segmentation based on colour pixel classification is largely unaffected by the choice of the colour space. However, segmentation performance degrades when only chrominance channels are used in classification tasks. In chrominance based methods, some valuable skin colour information will be lost whilst attempting to separate luminance from chrominance according to (Abdullah-Al-Wadud & Chae, 2008). Shin (Shin et al., 2002) question the benefit of colour transformation for skin tone detection, e.g., RGB and non-RGB colour spaces. Jayaram (Jayaram et al., 2004) conclude that the illumination component provides different levels of information on the separation of skin and non-skin colour, and thus the absence of illumination does not improve performance. This significant conclusion is drawn based on experiments on different colour transformations with and without illumination inclusion. Their data set comprises 850 images. Among those who incorporate illumination are (Lee & Lee, 2005), where they cluster human skin tone in the 3D space of YC_bC_r transformation. These authors are among those very few researchers who felt the exclusion of luminance is not preferred in the development of a skin tone classifier.

The proposed method goes a step further and shows that the abandoned luminance component carries considerable information on skin tone. The experiments herein lend some support to this hypothesis. Many colour spaces used for skin detection are simply linear transforms from RGB and as such share all the shortcomings of RGB (Ford & Roberts, 1998).

Probability-based classifiers have been developed to segregate skin tone regions such as the Bayes classifier used in (Liu & Wang, 2008). Additionally, (Liu & Wang, 2008) take advantage of inter-frame dependencies in video files. At first, the histogram of the skin pixels and non-skin pixels of the present frame is determined, then the conditional probability of each pixel belonging to the skin area and non-skin area is computed respectively. Next the ratio of these two conditional probabilities is computed. Finally, this ratio is compared with a threshold to determine its property as being a skin pixel or a non-skin pixel.

3.2.1 Orthogonal colour space (YC_bC_r)

The Y, C_b and C_r components refer to Luminance, Chromatic blue and Chromatic red respectively. This is a transformation that belongs to the family of television transmission colour spaces. This colour space is used extensively in video coding and compression, such as MPEG, and is perceptually uniform (Chi et al., 2006). Moreover, it provides an excellent space for luminance and chrominance separability (Beniak et al., 2008). *Y* is an additive combination of *R*, *G* and *B* components and hence preserves the high frequency image contents. The subtraction of *Y* in Equation (3.3) cancels out the high frequency (*Y*) (Lian et al., 2006). Given the triplet RGB, the YC_bC_r transformation can be calculated using the following system - Note: the transformation formula for this colour space depends on the used recommendation:

$$YC_bC_r : \begin{cases} Y = 0.299R + 0.587G + 0.114B \\ C_b = 0.56(B - Y) \\ C_r = 0.71(R - Y) \end{cases} \tag{3.3}$$

Hsu (Hsu et al., 2002) used C_bC_r for face detection in colour images. They developed a model where they noticed a concentration of human skin colour in C_bC_r space. These two components were calculated after performing a lighting compensation that used a “reference white” to normalise the colour appearance. They claimed that their algorithm detected fewer non-face pixels and more skin-tone facial pixels. Unfortunately, the testing experiments that were carried out using their algorithm were not in reasonable agreement with this assertion. Some of these results are shown here. Figure 3.2 describes the algorithm.

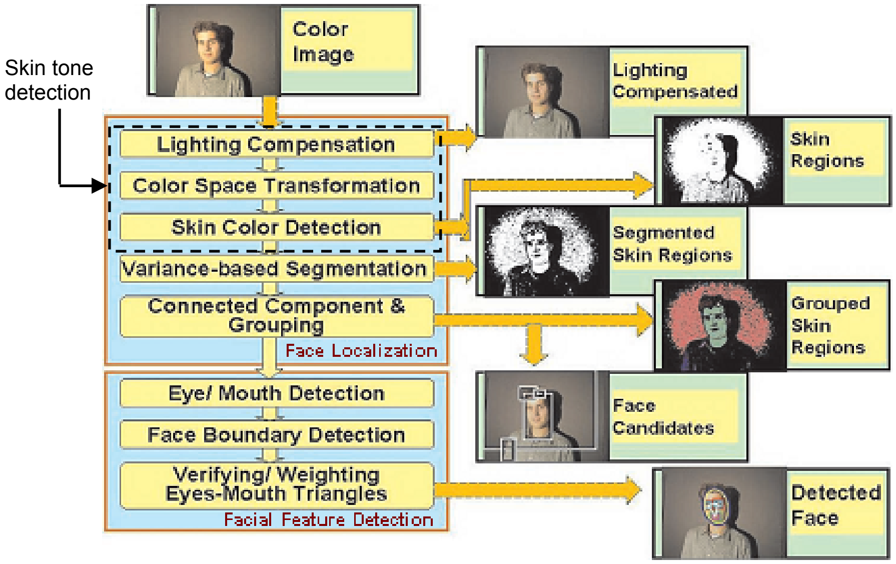


Figure 3.2: The system provided by Hsu (Hsu et al., 2002)

Similarly, Yun (Yun et al., 2007) used Hsu's algorithm with an extra morphological step where they propose a colour based face detection algorithm in the YC_bC_r colour space. The use of the illumination compensation method and a morphology closing was to overcome the difficulty of face detection applicable to video summary. Shin (Shin et al., 2002) showed that the use of such colour space gives better skin detection results compared to seven other colour transformations. The eight colours studied are: nRGB, normalized RGB, CIEXYZ, CIELAB, HSI, SCT, Spherical Coordinate Transform, YC_bC_r, YIQ, and YUV. RGB was used as a baseline performance. For each colour space they dropped its illumination component to form 2D colour.

Choudhury (Choudhury et al., 2008) develop a method tailored to fit, File Hound, which is a field analysis software used by law enforcement agencies during their forensic investigations to harvest any pornographic images from a hard drive. They propose a hybrid algorithm where the compound RGB and YC_bC_r based methods are exploited. They notice that the RGB method's disadvantage is compensated in YC_bC_r and vice versa. To this end, only relatively large regions which have been missed by the RGB filter are re-filtered through a YC_bC_r filter. Time complexity of their approach was not discussed.

Zhao (Zhao et al., 2008) construct a vector comprising of a blend of different selected components from different colour spaces of which C_r was present. Principal component analysis, PCA, was applied to this feature vector to find the main orthonormal axes which maximally de-correlate the sample data. A Mumford-Shah model was used to segment the image. All of the regions were then traversed to calculate the ratio of skin pixels to total pixels within each individual region. Only those regions whose ratio reaches the statistical value would then be regarded as skin regions. Their method entails off-line training and therefore its generalization is questioned.

Hsu's algorithm (Hsu et al., 2002) was chosen by Shaik and Asari (Shaik & Asari, 2007) to track faces of multiple people moving in a scene using Kalman filters. Zhang and Shi (Zhang & Shi, 2008) took the same approach with some modifications when the brightness of the face in an image was low. Their method is almost identical to (Hsu et al., 2002) except that they pre-process the image by setting all pixels below 80 to zero in all three primary colours, i.e., RGB. They claim their method works better under low brightness mainly due to the pre-processing phase. In order to avoid the extra computation required in conversion from RGB to HSV, Wong (Wong et al., 2003) use the YC_bC_r colour model and developed a metric that utilises all the components namely Y, C_b and C_r.

3.2.2 Log Opponent and HSV

The human visual system incorporates colour opponency and so there is a strong perceptual relevance in this colour space (Berens & Finlayson, 2000). The Log-Opponent, LO, uses the base 10 logarithm to convert RGB matrices into I, R_g, B_y as shown in Equation (3.4) - Note that this work does not assume a particular range for the RGB values:

$$IR_gB_y : \begin{cases} I = [L(R) + L(G) + L(B)]/3 \\ R_g = L(R) - L(G) \\ B_y = L(B) - [L(R) + L(G)]/2 \end{cases} \tag{3.4}$$

$$\text{where}, L(x) = 105 + \log_{10}(x+1).$$

This method uses what is called hybrid colour spaces. The fundamental concept behind hybrid colour spaces is to combine different colour components from different colour spaces to increase the efficiency of colour components to discriminate colour data. Also, the aim is to lessen the rate of correlation dependency between colour components (Forsyth & Fleck, 1999). Here, two spaces are used, namely log-opponent, IR_gB_y, and HS from the HSV colour space. HS can be obtained by applying a non-linear transformation to the RGB colour primaries as shown in Equation (3.5). A texture amplitude map is used to find regions of low texture information. The algorithm first locates images containing large areas where colour and texture is appropriate for skin, and then segregates those regions with little texture. The texture amplitude map is generated from the matrix I by applying 2D median filters. The RGB to HSV transform can be expressed as in Equation (3.5):

$$HSV : \begin{cases} H = \begin{cases} h, B \leq G \\ 2\pi - h, B > G \end{cases} \\ \text{where}, h = \cos^{-1} \dfrac{1/2[(R-G) + (R-B)]}{\sqrt{(R-G)^2 + (R-G)(G-B)}} \\ S = \dfrac{\max(R,G,B) - \min(R,G,B)}{\max(R,G,B)} \\ V = \max(R,G,B) \end{cases} \tag{3.5}$$

In order to segment potential face regions, Chen (Chen et al., 2008) analyze the colour of the pixels in RGB colour space to decrease the effect of illumination changes, and then classify the pixels into face-colour or non-face colour based on their hue, component *H* in Equation (3.5). The classification is performed using Bayesian decision rules. Their method degrades when the images contain complex backgrounds or uneven illumination.

3.2.3 Basic N-rules RGB (NRGB)

The basic N-rules RGB method is a simple yet powerful method to construct a skin classifier directly from the RGB composites which sets a number of rules, *N*, for skin colour likelihood. Kovač (Kovač et al., 2003) state that RGB components must not be close together, e.g., for luminance elimination. They utilize the following rules: An *R*, *G*, *B* pixel is classified as skin if and only if:

$$\begin{aligned} &R > 95 \;\&\; G > 40 \;\&\; B > 20 \\ &\&\; \max(R, G, B) - \min(R, G, B) > 15 \\ &\&\; |R-G| > 15 \;\&\; R > G \;\&\; R > B \end{aligned} \tag{3.6}$$

Some authors prefer to normalise the RGB primaries beforehand. Let the RGB denote the normalised colour space, which is expressed in Equation (3.7).

$$r = \frac{R}{R+G+B}, g = \frac{G}{R+G+B} \quad , b = \frac{B}{R+G+B} \tag{3.7}$$

The *b* component has the least representation of skin colour and therefore it is normally omitted in skin segmentation (Porle et al., 2007).

Abdullah-Al-Wadud and Chae (Abdullah-Al-Wadud & Chae, 2008) use a Colour Distance Map, CDM, applied to RGB colours, although this can be extended to any colour space. They implement an algorithm based on the property of the watershed to further refine the output using an edge operator. The generated CDM is a greyscale image. The distribution of the distance map is quasi-Gaussian in all cases. They also propose an adaptive Standard Skin Colour, SSC, to act as a classifier to vote for skin pixels. The method does not develop any colour space.

3.2.4 Other colour spaces

Porle (Porle et al., 2007) propose a Haar wavelet-based skin segmentation method in their aim to address the problem of extracting arms occluded in torsos in selected images. The segmentation procedure is performed using six different colour spaces, namely: RGB, rgb, HSI, TSL, SCT and CIELAB. They concluded that the *B* component, representing the position between yellow and blue, in the CIELAB colour space has the best performance. Obviously, this technique is complex and time consuming as it involves wavelets decomposition.

3.3 Summary

In this Chapter a review of image encryption and skin-tone detection algorithms has been given. The available algorithms in both disciplines have some drawbacks or inefficiencies. This has been discussed at length. In the case of image encryption, AES and Chaotic-based method suffer from having the avalanche property (discussed further on p. 130), additionally a balanced bit stream of 1s and 0s cannot be guaranteed. In the case of skin-tone detection, the negligence of luminance in most of the current methods has rendered those methods inefficient. Some of the algorithms in both disciplines suffer from slow execution.

The next chapter will describe in detail the proposed image steganography method, *Steganoflage*. It will discuss what has to be embedded, where it needs to be embedded and how it can be embedded.

CHAPTER
FOUR

Steganoflage: Object-Oriented Image Steganography

This chapter discusses the methodology of the proposed method and examines in detail the theoretical aspects of *Steganoflage*. It illustrates the proposed framework *Steganoflage* which links three multi-disciplinary components, encryption, skin-tone detection and steganography.

In this chapter, the concept of Object-Oriented Embedding, OOE, is introduced into information hiding in general and particularly to steganography. The algorithm takes advantage of computer vision to orient the embedding process. Although, any existing algorithm can benefit from this technique to enhance its performance against steganalysis attacks, this chapter also considers a new embedding algorithm in the wavelet domain using the Binary Reflected Gray Code, BRGC, instead of the conventional Pure Binary Code, PBC. In the realm of information hiding, some researchers focus on robustness, i.e., watermarking, while others focus on imperceptibility, i.e., steganography. This work advocates a new steganographic model that meets both robustness as well as imperceptibility.

Furthermore, the chapter proposes enhancing steganography using a new entity of security which encrypts the secret image prior to embedding it in the original image. Various hash algorithms are available such as MD5, Message Digest 5, and SHA-2, Secure Hash Algorithm, which hash data strings, thus changing their state from being in a natural state to a seemingly unnatural state. A hash function is more formally defined as the mapping of bit strings of an arbitrary finite length to strings of a fixed length

(Wang et al., 2008).

Here the aim is to extend SHA-2 to encrypt 2D digital data, the terminology and functions are described in the US Secure Hash Algorithm (SHA, 2001). The introduction of two transforms combined with the output of the SHA-2 algorithm creates a strong image encryption setting.

The proposed approach retains the structures readily available in the unencrypted bit stream. Such structures, often specified by special header patterns, would comply with standard multimedia codecs. Thus, an encrypted video for instance would still be successfully decoded. Other enhancements made are, the introduction of an object-based embedding by tracking skin tone areas and embedding using BRGC in the wavelet domain.

The following sections of this chapter discuss in detail the processes and stages of the *Steganoflage* algorithm. This commences with the description of a new encryption method designed for optical imagery. The three main sections, as shown in Figure 4.1, are stand-alone algorithms which can be applied in non-steganographic scenarios. However, here the algorithms are brought together and so they have a unique interaction. The first section describes what is to be embedded, section 4.1, where the embedding will occur, section 4.2, and finally how the embedding will occur, section 4.3. In each section of this chapter it is important to note the following points:

- The aim is to build a coherent algorithm from independent components assuring the overall optimality of the integrated algorithm
- The chapter describes the analytical formulation of each algorithm

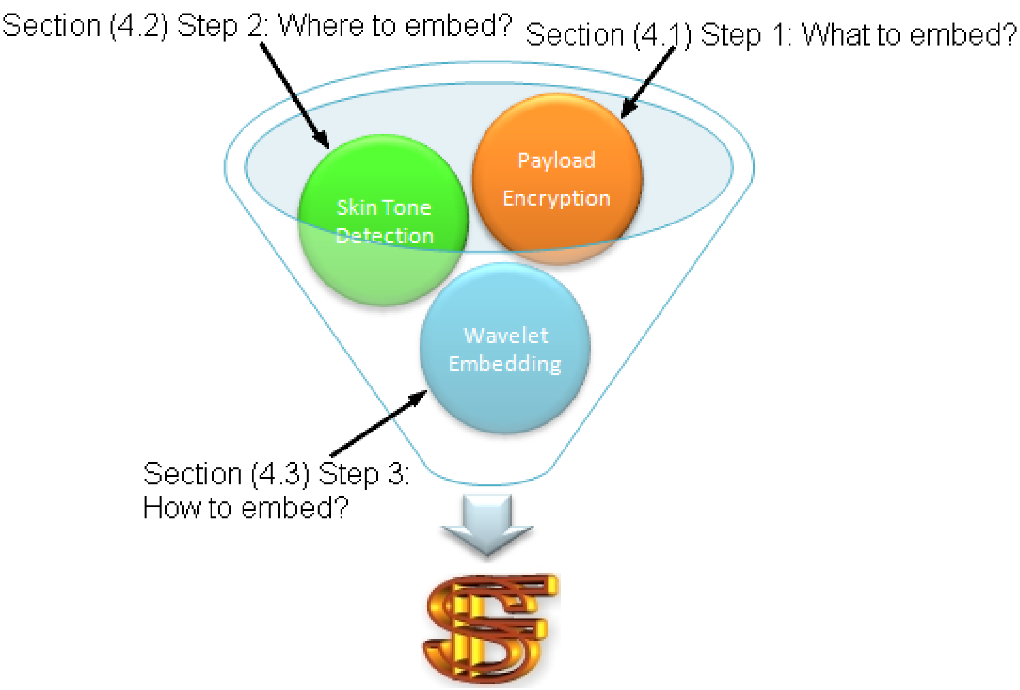

Section (4.2) Step 2: Where to embed?
Section (4.1) Step 1: What to embed?
Payload Encryption
Skin Tone Detection
Wavelet Embedding
Section (4.3) Step 3: How to embed?

Figure 4.1: The different components of the *Steganoflage* algorithm

4.1 Step 1: Payload Encryption (What to Embed?)

There exist several algorithms that deal with text encryption. However there has been little research carried out to date on encrypting digital images as was discussed in Chapter 3, section 3.1. This section describes a novel way of encrypting digital images with password protection using a 1D SHA-2 algorithm coupled with a compound forward transform as shown in the code in Figure B.1 (Appendix). A spatial mask is generated from the frequency domain by taking advantage of the conjugate symmetry of the complex imagery part of the Fourier Transform. This mask is then XORed with the bit stream of the original image. Exclusive OR, a logical symmetric operation, yields 0 if both binary pixels are zeros or if both are ones and 1 otherwise. This can be verified simply by modulus (pixel1, pixel2, 2). Finally, diffusion is applied based on the displacement of the cipher's pixels in accordance with a reference mask. This process yields an encrypted version used as a payload. One of the merits of such an algorithm is to force a continuous tone payload to map onto a balanced bits distribution sequence where the number of {1} bits is equal to the number of {0} bits. This bit balance is needed in steganographic applications as it is likely to have a balanced perceptibility effect on the cover image when embedding.

4.1.1 A new image encryption algorithm

This proposal exploits the strength of a 1D hash algorithm, SHA-2, and extends it to handle 2D data such as images. SHA functions “are highly flexible primitives that can be used to obtain privacy, integrity and authenticity” (Denis, 2006). The DCT and FFT are incorporated into the process to increase the disguise level and thus generate a random-like output that does not leave any distinguishable pattern of the original image. The ordering of the transforms is crucial since the algorithm’s strength is attributed to exploiting the symmetrical property of the FFT’s imaginary part.

The exhaustive step-by-step description of the encryption algorithm is illustrated in Figure 4.2. The method works as a one-time pad cipher in which the extended key is used only once, therefore, the decryption will follow the same digital process but with the cipher input into *Steganoflage*, i.e., symmetric encryption. Starting with a password phrase *K* supplied by the user the algorithm generates a SHA-2, i.e., SHA-256, based hash string *H (K)* which forms the initial condition. The vector *H*, treated as a string of hexadecimal characters, is then converted to its decimal version and finally transformed to a bit stream matrix of fixed dimension [8x32]. Parallel to this, the original image A is converted to a bit stream and reshaped to the order $8xMN$.

The partially extended key, herein *K’*, is still short to accommodate the image bit stream. Therefore, the algorithm performs key full expansion towards the needed dimension, herein $8xMN$. Obviously, this step would result in repetitive patterns that would make the ciphered image prone to attacks, a problem that was independently noticed in (Usman et al., 2007). To alleviate this problem the method applies a thresholded DCT, where Equation (4.1) is used, followed by a FFT to provide the confusion requirement and to tighten the security. Note that nested transforms are commonly found in the literature, for example O'Ruanaidh and Pun (O'Ruanaidh & Pun, 1997) used a FFT followed by log-polar mapping and a FFT to embed a watermark.

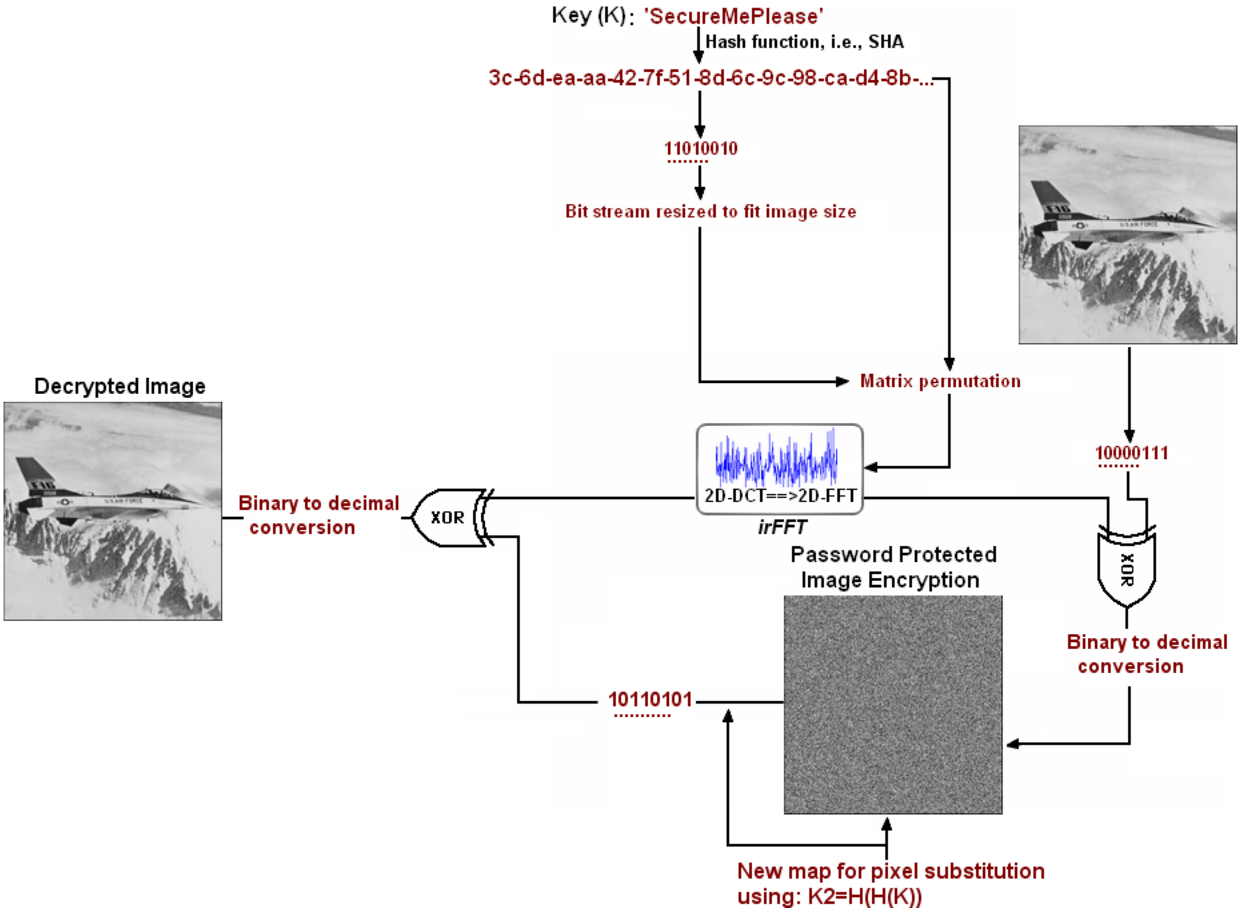

Key (K): 'SecureMePlease'
Hash function, i.e., SHA
3c-6d-ea-aa-42-7f-51-8d-6c-9c-98-ca-d4-8b-...
11010010
Bit stream resized to fit image size
Matrix permutation
Decrypted Image
10000111
2D-DCT==>2D-FFT
irFFT
Binary to decimal conversion
XOR
Password Protected Image Encryption
XOR
Binary to decimal conversion
10110101
New map for pixel substitution using: K2=H(H(K))

Figure 4.2: Block diagram of the proposed image encryption algorithm

$$f(u,v) = \frac{1}{8MN}\sum_{x=0}^{7}\sum_{y=0}^{MN-1} F(x,y)e^{-2\pi i(xu/8+yv/MN)} \qquad (4.1)$$

$$, satisfying \quad Equation \quad (4.2)$$

$$where, F(x,y) = DCT(\lambda_{8,MN}), subject \quad to \quad :$$

$$\mathrm{F}(\mathrm{x},\mathrm{y}) = \begin{cases} 1 & \text{iff} \quad \mathrm{DCT}(\lambda_{8,\mathrm{MN}}) > 0 \\ 0 & \text{Otherwise} \end{cases}$$

Here $\lambda_{8,MN}$ denotes the resized key where the subscripts *M* and *N* denote the width and height dimensions of the image, respectively. The FFT operates on the DCT transform of $\lambda_{8,MN}$ subject to Equation (4.2).

Generating a pseudo-random binary sequence from the orbit of $f(u,v)$ requires the mapping of the state of the system to its binary values $\{0,1\}$. One clear method for converting a real number to a discrete bit symbol is to use a rule as shown in Equation (4.2). Given the output of Equation (4.1) the corresponding binary map can be derived:

$$\mathrm{Map}(x,y) = \begin{cases} 1 & \text{iff} \quad \mathrm{imag}(f(u,v)) > \mathrm{thr} \\ 0 & \quad\text{Otherwise} \end{cases} \tag{4.2}$$

where thr is an appropriately selected threshold value and *imag(•)* denotes the imaginary part of the complex function which can be compared directly with a threshold thr. For a balanced binary sequence and for robustness, *thr* should be chosen such that the probability *P* $(\mathrm{imag}(f(u,v)) < \mathrm{thr})$ = *P* $(\mathrm{imag}(f(u,v)) > \mathrm{thr})$. Fortunately, the imaginary part of the signal $f(u,v)$ is always symmetrical around zero, see Chapter 6 for validity of this property. Therefore, $\mathrm{thr} = 0$ is an explicit solution. Since the coefficients in this calculation are converted to a binary map the reverse construction of the password phrase is impossible. Hence the name Irreversible Fast Fourier Transform, *IrFFT*.

The generated bit-pattern exhibits sufficient randomness to provide cryptographic security as shown in Chapter 6. This map finally is XORed with the bit stream version of the image. The result is then converted into greyscale and reshaped to form the ciphered image. The coding phase uses the Map, as shown in Equation (4.3), to encrypt the bit stream of image *A* and produce a new encrypted matrix A', in such a way that:

$$\varepsilon_{auth} \equiv \{A - D(A',\mathrm{Map})\} \tag{4.3}$$

where, $D(A',\mathrm{Map})$ denotes the decoding of A' with the same key generated Map. Ideally, ε_{auth} should be equal to {Ø}, the null set, and starts to deviate from that when A' undergoes an image processing attack.

Another phenomenon which has been exploited was the sensitivity of the spread of the FFT coefficients to changes in the spatial domain. Therefore when this is coupled with the sensitivity of the SHA-2 algorithm to changes of the initial condition, i.e., password phrase, the Shannon law requirements can be easily met. For instance slight changes in the password phrase will, with overwhelming probability, result in a completely different hash and therefore a completely different Map.

The core idea here is to transform this sensitivity into the spatial domain where the 2D-DCT and the 2D-FFT can be applied to introduce sensitivity into the two dimensional space. As such, images can be easily encoded securely with password protection. Note that this scheme efficiently encrypts greyscale and binary images. However, for RGB images it is noticed that using the same password for the three primaries yields some traceable patterns inherited from the original image, RGB colours are highly correlated. This is easily overcome through the following two choices: either the user supplies three passwords each of which encrypts one colour channel or more conveniently *Steganoflage* generates another two unique keys from the original supplied password. For instance, a single key can be utilized to generate the following different hash functions $H(\overrightarrow{K}), H(\overleftarrow{K})$, and $H(H(\overrightarrow{K}))$ to encrypt the R, G and B channels, respectively. K denotes the supplied key, the arrows indicate the string reading directions and $H(H(\bullet))$ denotes double hashing.

There are many applications for this extended 2D SHA-2 algorithm, however this thesis concentrates solely on the strengthening of digital image steganography. The function used for encryption is shown in Figure A.1, see Appendix A.

4.2. Step 2: Identifying Embedding Regions (Where to Embed?)

This step discusses the automatic identification of reliable regions in images to serve for orienting the embedding process.

Illumination is evenly smeared along RGB colours in any given colour image. Hence, its effect is scarcely distinguished here. There are different approaches to segregate such illumination. The transformation matrix used here is defined in Equation (4.4).

$$\vec{\alpha} = [0.29893602\ 1293775390\ ,\quad 0.58704307\ 4451121360\ ,\quad 0.14020904\ 2551032500\ ]^{T} \tag{4.4}$$

where the superscript *T* denotes the transpose operator to allow for matrix multiplication.

Let Ψ denote the 3D matrix containing the RGB data of the host image with the width (*W*) and height (*H*), and let $x \in [1,2,...,n]$, where $n = W \times H$. Note that this method acts on the RGB colours stored in double precision, i.e., linearly scaled to the interval [0 1]. The initial colour transformation is given in Equation (4.5).

$$I(x) = \left(\Psi(r(x), g(x), b(x)) \otimes \vec{\alpha}\right) \tag{4.5}$$

where $\otimes$ represents matrix multiplication. This reduces the RGB colour representation from 3D to 1D space. The vector *I(x)* eliminates the hue and saturation information whilst retaining the luminance. It is therefore regarded formally as a greyscale colour.

Next, the algorithm tries to obtain another version of the luminance but this time without taking the *R* vector into account. Most skin colour tends to cluster in the red channel. The discarding of the colour red is deliberate, as in the final stage it will help to calculate the error signal. Therefore, the new vector will have the largest elements taken from *G* or *B*:

$$\hat{I}(x) = \max_{x \in \{1,...,n\}}(G(x), B(x)) \tag{4.6}$$

Equation (4.6) is actually a modification of the way HSV computes the *V* values. The only difference is that the method does not include in this case the red component in the calculation. Then for any value of *x*, the error signal is derived from the calculation of element-wise subtraction of the matrices generated by Equation (4.5) and Equation (4.6) which can be defined as given in Equation (4.7).

$$e(x) = I(x) - \hat{I}(x) \tag{4.7}$$

Note that $e(x)$ must employ neither truncation nor rounding.

Creating a Skin Probability Map, SPM, that uses an explicit threshold based skin cluster classifier which defines the lower and upper boundaries of the skin cluster is crucial to the success of the proposed technique. A collection of 147852 pixel samples was gathered from different skin regions exhibiting a range of races with extreme variation of lighting effect. After transformation using the proposed method, the projection of data admits a distribution that could easily fit a Gaussian curve using an Expectation Maximization, EM, method which is an approximation of Gaussian Mixture Models, GMM, as shown in Figure 4.3. It is also clear that there are no other Gaussians hidden in the distribution. To identify the boundaries, some statistics need to be computed.

Let μ and σ denote the mean and standard deviation of the above distribution, and let Δ_{left} and Δ_{right} denote the distances from μ on the left and right sides respectively. The boundaries are determined based on Equation (4.8).

$$\begin{aligned} &\mu - (\Delta_{\text{left}} * \sigma) \approx 0.02511 \\ &\mu + (\Delta_{\text{right}} * \sigma) \approx 0.1177 \end{aligned} \tag{4.8}$$

Here Δ_{left} and Δ_{right} are chosen to be one and three sigma away from μ respectively to cover the majority of the area under the curve. Hence, the precise empirical rule set for this work is given in Equation (4.9), which is a function $f : \chi \to \{0,1\}$ such that:

$$f_{\text{skin}}(x) = \begin{cases} 1 \text{ if} \quad 0.02511 <= e(x) <= 0.1177 \\ 0 \qquad\qquad \text{otherwise.} \end{cases} \tag{4.9}$$

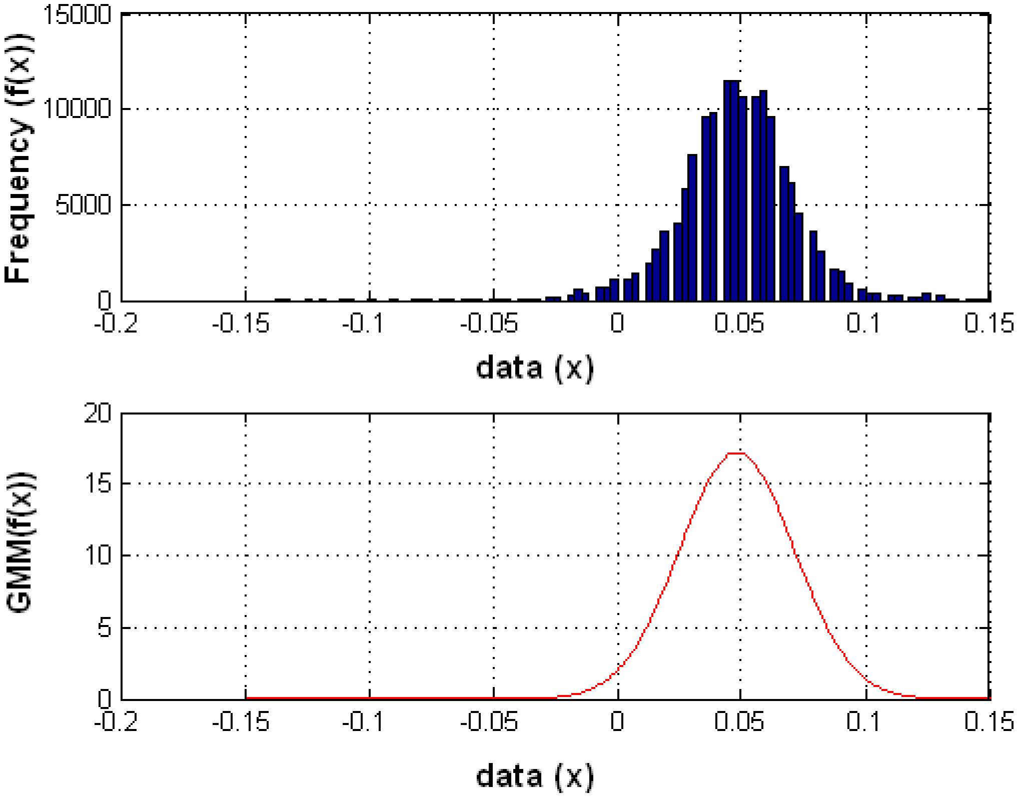


Figure 4.3: Frequency distribution of the data (top) and its Gaussian curve fit (bottom)

This work claims that, based on extensive experimentation, this rule pins down the optimum balanced solution. Even though the inclusion of luminance was adopted, the 3D projection of the three matrices $I(x), \hat{I}(x), e(x)$ shows clearly that the skin tone clusters around the boundaries given in Equation (4.8). This is shown in Figure 4.4. Notice how compact the skin tone is, using the proposed method. This practical example contradicts the claim reported previously in (Hsu et al., 2002) showing the deficiency of using luminance in modelling skin tone colour. The hypothesis that this work supports is that luminance inclusion does increase separability of skin and non-skin clusters. In order to provide evidence for this hypothesis, the proposed algorithm was tested on different RGB images with different background and foreground complexities. Some images are selected which expose uneven transitions in illumination to demonstrate the robustness of the skin tone algorithm.

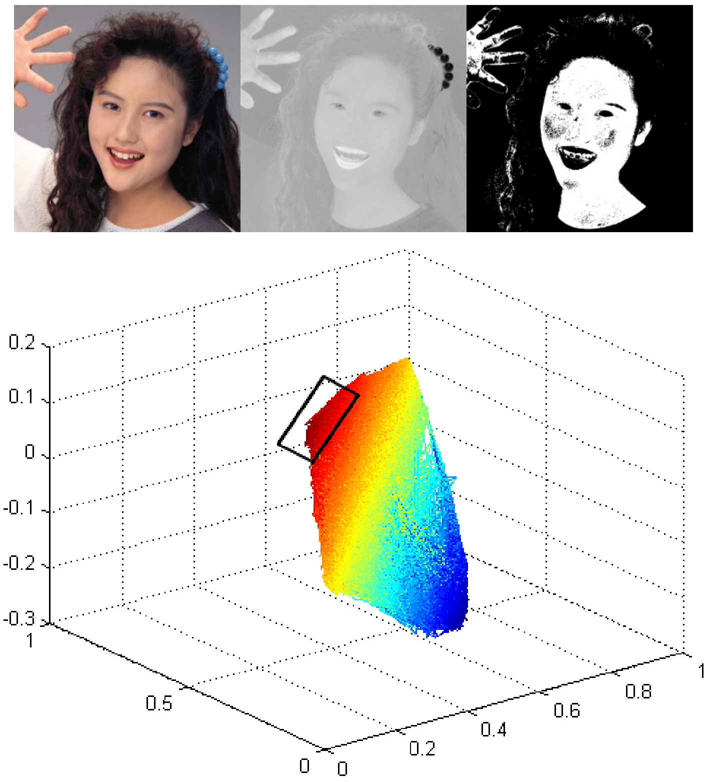


Figure 4.4: Skin tone segmentation using the proposed method. The figure illustrates: (top and left to right) original image, result of applying Equation (4.7), result of applying Equation (4.9), and skin tone cluster in a 3D mesh, respectively. The dark red dot cloud represents the region where skin colour tends to cluster, i.e., the area bounded by a rectangle

For greyscale face images, the algorithm described in (Cheddad et al., 2008b) can be used, which has the advantage of ease of implementation. Given a set of 2D points, the Voronoi region for a point P_i is defined as the set of all the points that are closer to P_i than to any other points. That can be formulated more formally: Let $S = \{P_1, P_2 \ldots P_n\}$ be a finite subset of R_m and let $d: R_m \times R_m \rightarrow R$ be a metric. The Voronoi region is defined as $VR(P_i)$ of a point P_i via $VR(P_i) = \{P \in R_m \mid d(P, P_i) \leq d(P, P_j) \text{ for all } j = 1, 2, \ldots, n, j \neq i\}$, i.e., $VR(P_i)$ is the set of all points that are at least as close to P_i as to any other point of S. The set of all 'n' VR is called the Voronoi Diagram VD(S) of S (Costa & Cesar, 2001). VD is generated from a set of sites that correspond to the image histogram bin values. In essence, these points are ≤ 255. A set of triangulation vertices is then produced, known as a Delaunay Triangulation, which dictates the graph cut. After image segmentation template matching is used to vote for a face blob as shown in Figure 4.5.

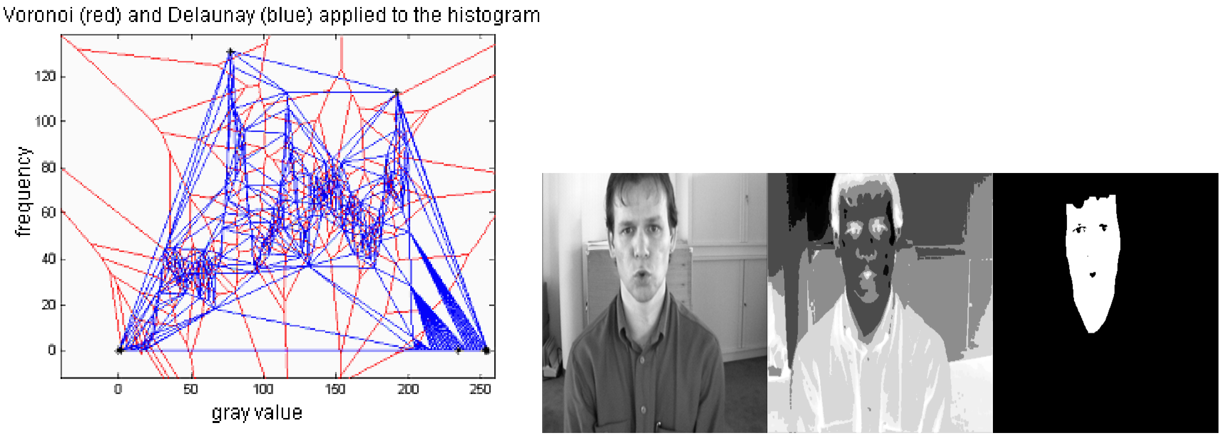


Figure 4.5: Test result of face segmentation in gray scale. The figure shows: (left to right) Delaunay Triangulation generated from points of the histogram, original image, Voronoi based image segmentation and segmented face, respectively.

4.3 Step 3: The Embedding (How to Embed?)

After generating the encrypted payload, the colour transformation $RGB \rightarrow YC_bC_r$ is applied to the cover image that carries the encrypted data. Also, the use of such a transformation segments chromatic homogeneous objects in the cover image, i.e., human skin regions. The YC_bC_r space can remove the strong correlation among *R*, *G*, and *B* matrices in a given image.

The majority of the introduced steganographic techniques suffer from intolerance to geometric distortions applied to the stego-image. For instance, if rotation or translation occurs all of the hidden data will be lost. A solution to this problem could be through incorporating computer vision into the process. The concept of *OOE* now becomes one of finding clusters of skin areas in the image 2D space. The algorithm starts by first segmenting probable human skin regions such that:

$$C = C_{bg} \cup C_{fg}, \text{where}, C_{fg} \in \{\bigcup_{i=1}^{n} S_i\}, S_i \cap S_j = \emptyset, \forall i \neq j \tag{4.10}$$

In Equation (4.10) C, C_{bg}, and C_{fg} denote the cover image, the background regions and the foreground regions respectively. $\varnothing$ denotes the empty set and (S_1, S_2,…, S_n) are connected subsets that correspond to skin regions.

Based on experimentation, it is found that embedding into these regions produces less distortion to the carrier image compared to embedding in a sequential order or in any other areas. Such phenomena result from the fact that the eye does not respond with equal weight of sensitivity to all visual information. This is consistent with the claim that certain information simply has less relative importance than other information in the human visual system (Gonzalez & Woods, 2002). This information is said to be psycho-visually redundant since it can be altered without significantly impairing the quality of the image perception (Gonzalez & Woods, 2002). Human presence in digital photography and video files encourages such an approach. In this context, the postulation of the above skin model would definitely help in the case of image translation as it is invariant to such distortions. With reference to Equation (4.11), if the cover image is geometrically transformed by a translation of t_x, along the x axis, and t_y, along the y axis, in such a way that the new coordinates are given by:

$$C'\begin{bmatrix} x' \\ y' \end{bmatrix} = C\begin{bmatrix} x + t_x \\ y + t_y \end{bmatrix} \tag{4.11}$$

then each detected skin blob will be transformed likewise with the same distance to the origin as shown in Equation (4.12).

$$S_i'\begin{bmatrix} x' \\ y' \end{bmatrix} = S_i\begin{bmatrix} x + t_x \\ y + t_y \end{bmatrix}, \ \forall i \in \{1,\ldots,n\} \tag{4.12}$$

Skin regions are extracted based on colour tone, therefore, are undisturbed by translation (Cheddad et al., 2008e) and (Cheddad et al., 2009a).

To cope with rotation, it is sufficient to locate face features, i.e., eyes, based on the method described in (Zhao et al., 2008). Salient features form reference points that dictate the orientation of embedding and thus aid recovery from rotational distortions, see, Figure 4.6. Other types of attack are shown in Figure 4.7. The figure depicts: (left) shows the original cover image -ID01_035.bmp- obtained from GTAV Face Database

(Tarrés & Rama, n.d.) along with the image annotation to embed, (middle) attacked stego-image with half transparent frame and the extracted annotation and finally (right) shows an attack on stego-image with translation to the left with an offset=200 pixel and the extracted annotation which is identical to the embedded one.

Rotation about the origin is defined as in Equation (4.13).

$$x' = x\cos\theta - y\sin\theta, \quad (4.13)$$
$$y' = x\sin\theta + y\cos\theta$$

The angle θ is determined from the above elliptical model. Hence, if the attacked image is rotated in the opposite direction with the same angle, i.e., $\theta' = -\Delta\theta$ caused by the attack, the method will be able to restore the angle and will have the coordinates as shown in Equation (4.14).

$$x' = x, \quad (4.14)$$
$$y' = y$$

Equation (4.14) is used where embedding occurs in the neutralised orientation where $b_{\text{axis}} \perp x_{\text{baseline}}$. However, the encoder has 359 choices for the angle as expressed in Equation (4.15).

$$\theta' = \theta \pm \alpha \quad (4.15)$$

where $\alpha \in \{1,2,...,359°\}$ and denotes an agreed upon scalar which can form another optional secret key. Note that, for simplicity, α here belongs to the discrete space while in practice it is continuous. However the use of discrete values is encouraged in order to minimise the errors in the recovered bits.

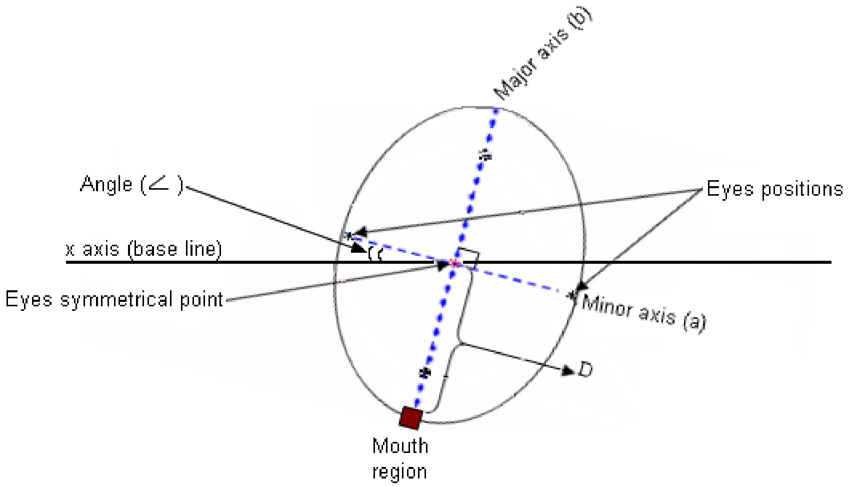


Figure 4.6: The elliptical model formed by face features

In addition to this, the algorithm yields a robust output against reasonable noise attacks and translation. Robustness against noise is due to the embedding in the 1st-level 2D Haar DWT with the symmetric-padding mode. DWT is a well known transformation that gained popularity among the image processing community especially those working in the area of image compression. Its applications in different areas is growing however, note that JPEG2000 uses DWT to compress images.

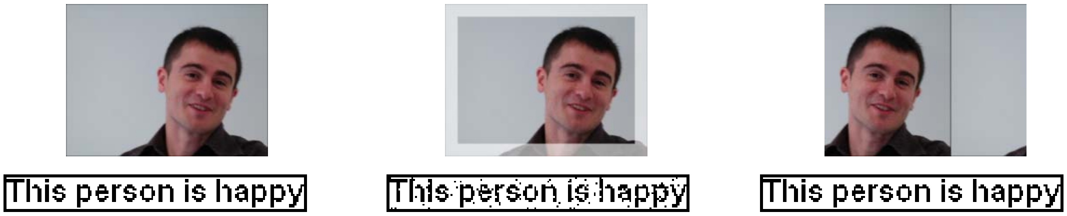


Figure 4.7: Resistance to other deliberate image processing attacks. (Top) stego-images and (bottom) extracted hidden data

RGB primary colours, as discussed earlier, contain a mixture of chromatic and luminance components. Moreover, the correlation between the three matrices is high. This is the reason why JPEG compression uses the $RGB \rightarrow YC_bC_r$ transformation as a pre-processing step. Algorithms based on DWT experience some data loss since the reverse transform truncates the values if they are saturated (go beyond the lower and upper boundaries, i.e., [0 255], and also for the round-off issue. In the process of identifying which colour transform to use, it is noted that other transforms, such as HSV and NTSC colour spaces, result in float double precision values that make steganography implementations difficult.

YC_bC_r is the chosen colour transform. The three channels have a different perceptual weighting and allow an embedding adjusted to the human visual perception (Rosenbaum & Schumann, 2000). Human skin tone tends to have a distinguishable dense presence along the middle range in the C_r component, which allows for coefficient modifications without impairing the visual quality, however, it distorts the first order statistics and does not resist compression. The luminance Y is a good compromise. It should be noted that embedding into any of these components would result in changes being made to all RGB corresponding values. In other words, the impact of embedding will be spread among RGB colours due to the very nature of the inverse transform, i.e., $\mathrm{YC_bC_r} \rightarrow \mathrm{RGB}$. Also, knowing that human skin tone resides along the middle range of the YC_bC_r different components allows the embedding into the DWT of the Y/C_r channels without introducing truncation issues. The coefficients of the DWT in the skin tone areas guarantee having relatively big amplitude signals which have strong noise immunity (Chen, 2007). Finally, this mechanism would leave the perceptibility of the stego-image virtually unaffected since the changes made in this transform will be spread among the RGB colours when inverse transformed.

Wavelet is chosen rather than DCT for the following reasons:

- the wavelets transform models better the Human Vision System, HVS, more closely than DCT does
- visual artefacts introduced by wavelet coded images are less evident compared to DCT because the wavelet transform does not decompose the image into blocks for processing

In addition to the above, the DFT, Discrete Fourier Transform, and the DCT are full frame transforms. Hence any change in the transform coefficients affects the entire image except if DCT is implemented using a block based approach. However DWT has spatial frequency locality, which means if the signal is embedded it will affect the image locally (Potdar et al., 2005a). Thus a wavelets transform provides both frequency and spatial descriptions for an image. More helpful to information hiding, the wavelet transform clearly separates high-frequency and low-frequency information on a pixel-by-pixel basis (Raja et al., 2006). Additional verification can be found in (Silva & Agaian, 2004).

For binary stream processing, there are two methods for converting decimal integers to a binary string. One is to use the conventional decimal to binary conversion called *PBC* and the other is termed the *BRGC* (WolframMathWorld, 1999). This binary mapping is the key to the augmented embedding capacity introduced by the method named “A Block Complexity Data Embedding, ABCDE” proposed in (Hioki, 2002). There is a trade-off, however, between robustness and distortion.

The central focus of this thesis is to embed the secret message into the approximation decomposition in the first-level 2D Haar DWT with the symmetric-padding mode guided by the detected skin tone areas. A coefficient’s precision is left intact while only its integer element carries the secret bit using BRGC. In PBC, the last LSB where the steg-value, compared to the plain-value, is unchanged, increased or decreased by one, i.e., change by ± 1 in the 1st LSB or ± 4 in the 3rd LSB, eventually leaves traceable statistical violations. Many algorithms to date still use such conventional models either in the spatial domain or the transform domain.

The BRGC allows alteration to even the third LSB, i.e., change by ± 3, in the DWT without much degradation compared to the conventional use of PBC. Figure 4.8 depicts the graphical structure of both methods. Let a plain-image pixel at the approximation level of a 1st level DWT be the coefficient *C* and let the secret bit be ‘0’: C=325.09821988712. Therefore:

- BRGC

C_{int}=325, Store=.09821988712

BRGC (C_{int}) = ‘111100111’

Stego-image (BRGC (C_{int})) =‘111100011’

BRGC -to-Decimal=‘111100011’➔322

Stego-image=Concatenate (322, Store) = 322.09821988712

Difference $\pm$ 3, odd number.

- PBC

Bin (C_{int}) = $(101000101)_2$

Stego-image (Bin (C_{int})) = $(101000001)_2$

Bin-to-Decimal = $(101000001)_2$➔321

Stego-image= Concatenate (321, Store) = 321. 09821988712

Difference $\pm$ 4, even number.

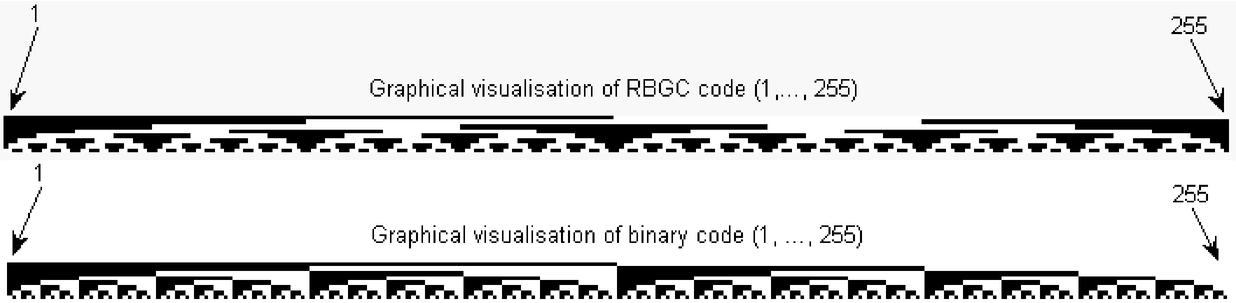

1
255
Graphical visualisation of RBGC code (1,..., 255)
1
255
Graphical visualisation of binary code (1, ..., 255)

Figure 4.8: RBGC and PBC contrast in the graphical space

The resistance to geometric distortions is feasible since, unlike S-Tools and F5 algorithms discussed in Ch.2, when skin tone blobs are selected then eye coordinates can be detected which act as reference points to recover the initial orientation. This makes the method immune to both rotation and translation.

The proposed encryption scheme was applied to digital image steganography for three reasons:

- embedding a random-like data into the Least Significant Bits, LSBs, would perform better than embedding the natural continuous-tone data

- for security and fidelity reasons the embedded data must undergo a strong encryption so even if it is accidentally discovered, which is unlikely to happen, the actual embedded data would not be revealed
- the stego-image may encounter some noise inference or geometric distortion which could change its intensity values and essentially the hidden encrypted data, therefore the encryption algorithm must be flexible enough to reconstruct the plain data.

Next, the different steps to construct the embedding algorithm are highlighted. Let *C* and *P* be the cover image and the payload respectively. The stego-image *S* can be obtained by the following embedding procedure:

Step 1: Encrypt *P* using the proposed encryption method to find *P'*

Step 2: Generate skin tone map, skin_map, from the cover *C* and determine, if desired, the agreed-upon orientation for embedding using face features as described earlier, embedding angle will be treated as an additional secret key. This goes along with the *Kerchoff*'s principle that states the security of an algorithm, which is assumed to be made public, resides in the secret key.

Step 3: Transform *C* to YC_bC_r color space

Step 4: Decompose the channel *Y* by one level of 2D-DWT to yield four sub-images (CA, CH, CV, CD)

Step 5: Resize skin_map to fit CA

Step 6: Convert the integer part of coefficients of CA into BRGC code and store the decimal values

Step 7: Embed, the embedding location of data is also randomized using the same encryption key, the bit stream of *P'* into the coefficients' BRGC of the skin area in CA guided by the skin_map. The coefficient's 3rd least significant bit is chosen to embed the secret bit while random bits are embedded simultaneously into the coefficient's 1st and 2nd least significant bit. This procedure is known as masking and it helps overcome few compression errors

Step 8: Convert the modified BRGC code back to coefficients, restore the decimal precision and reconstruct *Y'*

Step 9: Convert Y'C_bC_r to RGB colour space and obtain the stego-image, i.e., *S*. Note that the effect of embedding is spread among the three channels RGB since the due to such conversion.

The decoding stage essentially follows steps 2-6 while step 7 refers instead to the extraction phase of the secret bits. Then the decryption of the bit stream will be carried out. These steps are expressed graphically in Figure 4.9. Embedding into the 'Y' channel has the advantage of better resistance to compression, while embedding into the 'C_r' channel has the advantage of better image perceptibility at the expense of resistance to image compression.

Figure 4.10 shows an example of the test data with the PSNR. Note the use of biometric facilitates having the embedding invariant to rotation and translation. The figure shows: (left) the payload, herein CT scan of a young female (Scottish Radiological Society, 2002) and its encrypted version, each shown with their respective histogram, notice how the encryption gives all the gray values almost equal probability of occurrence, (right) concealment of the encrypted medical data in an innocuous face image.

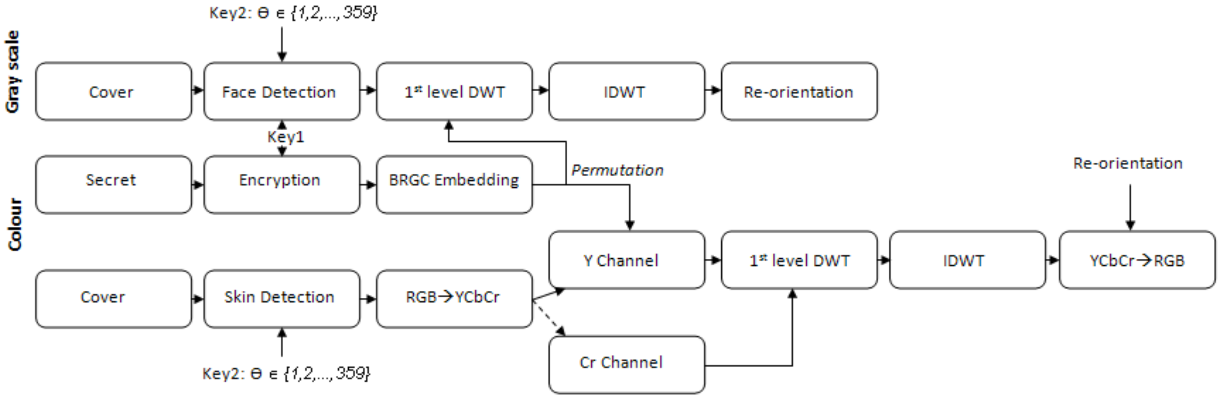


Figure 4.9: Block diagram of the proposed steganography method

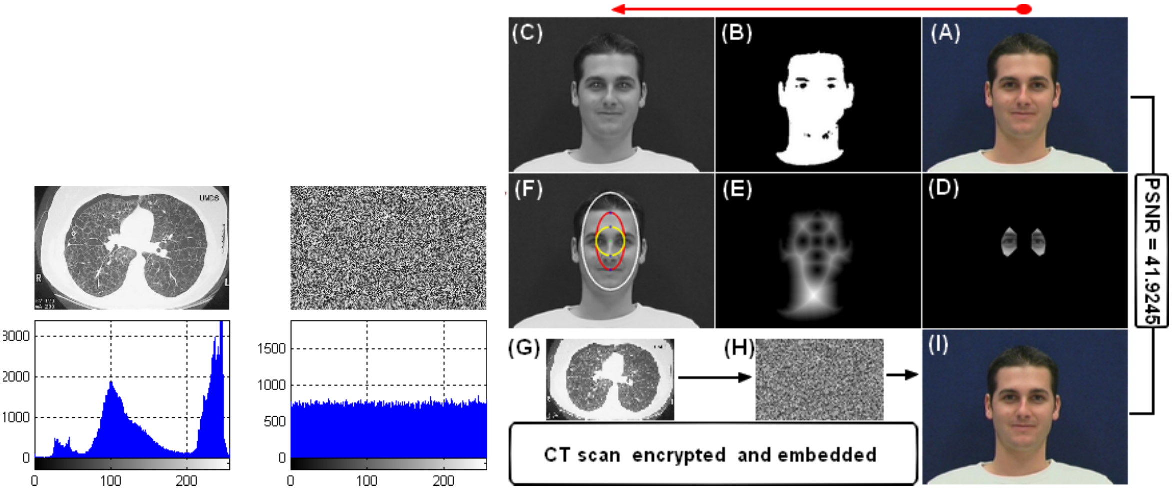


Figure 4.10: The proposed *Steganoflage.* (Left) encrypted data with histograms and (right) the embedding process using face features

4.4 Summary

This chapter proposes an object-oriented embedding approach to steganography, it is possible thanks to established computer vision algorithms. It also becomes apparent the different advantages that the discussed new algorithms for image encryption and the real-time skin tone detection can bring to the area of steganography. In summary, this chapter has examined in detail the main components that define the proposed algorithm.

Extending this method to video files would solve the problem of the limited payload available by targeting skin regions. However, a steganographer may choose to consider the entire image for embedding, and then detecting skin area would reduce to just providing the desired secret embedding angle. Nevertheless, the proposed scheme has some advantages. For example identifying skin areas will give an instant direct split of an image into two main areas, one for embedding and another to correct for any statistical distortion caused.

The proposed encryption method as well as the skin-tone detection algorithm can be used in other related disciplines.

In the next chapter, Chapter 4, an insight into the development of *Steganoflage* is given. Chapter 5 analysis and evaluates each component of *Steganoflage*, image encryption and skin-tone detection, and tests its overall robustness.

CHAPTER

FIVE

Implementation of Steganoflage

This chapter discusses the different phases of implementation of *Steganoflage*. It also explores a unique injection of HTML, Hyper Text Markup Language, JavaScript or PHP, a hypertext pre-processor, codes in MATLAB internal scripts which allow MATLAB to communicate with the browser flexibly. *Steganoflage* has offline and online interfaces. The chapter also discusses applications of *Steganoflage*.

5.1 Development Environment

Steganoflage is an integrated system built on MATLAB scripts, HTML and PHP. MATLAB is a high-performance language integrating computation, visualization, and programming in an easy-to-use environment. PHP is a server-side HTML embedded scripting language. It is designed for building dynamic websites, with Apache serving as server engine. PHP usually comes with a lightweight very fast database system called MySQL, a Structural Query Language, which supports RDBMS, Relational Database Management System. PHP, Apache and MySQL are all open source software.

5.2 Architecture of *Steganoflage*

Steganoflage consists of core modules to allow easy visualization of its inner architecture. The key steps that were involved in the building of *Steganoflage* were:

- Defining the goals of *Steganoflage*, i.e., implementing a complete steganographic system with tailored encryption and pattern recognition pre-processing stages.
- Defining the project's scope and objectives which stem from Chapter 1.

- Modularisation in terms of assembling *Steganoflage* where different modules interact with one another as shown in
- Figure **5.1**.
- The encoding and decoding modules are linked to two sub-routines, encryption and skin-tone detection. User interaction can either be offline or online.
- Formation of modules' creation to realize the full initial specification.
- System testing against attacks scenarios. This involves initially testing each module against a range of attacks. Subsequently, a final test was carried out to illustrate the efficiency and security of the overall system *Steganoflage*.
- Adjusting *Steganoflage* by fixing bugs, adjusting inner parameters adjustments and making further adjustments to improve human computer interaction aspects of *Steganoflage*.

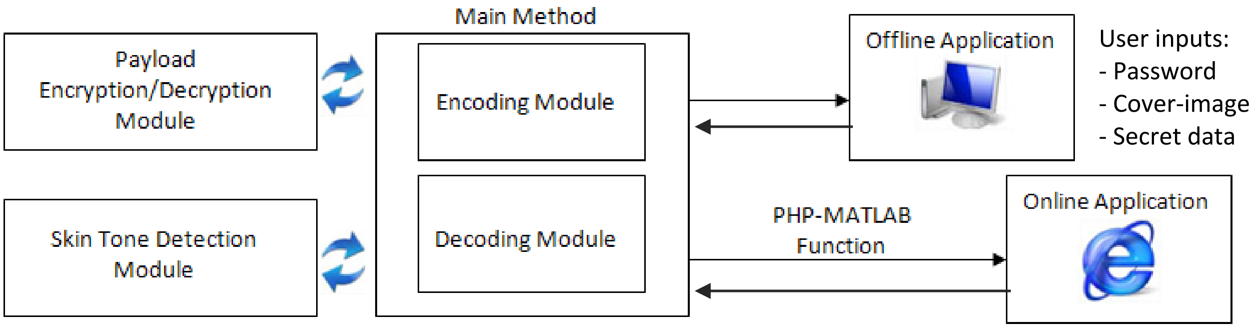


Figure 5.1: Generic Architecture of *Steganoflage* showing offline and online interfaces

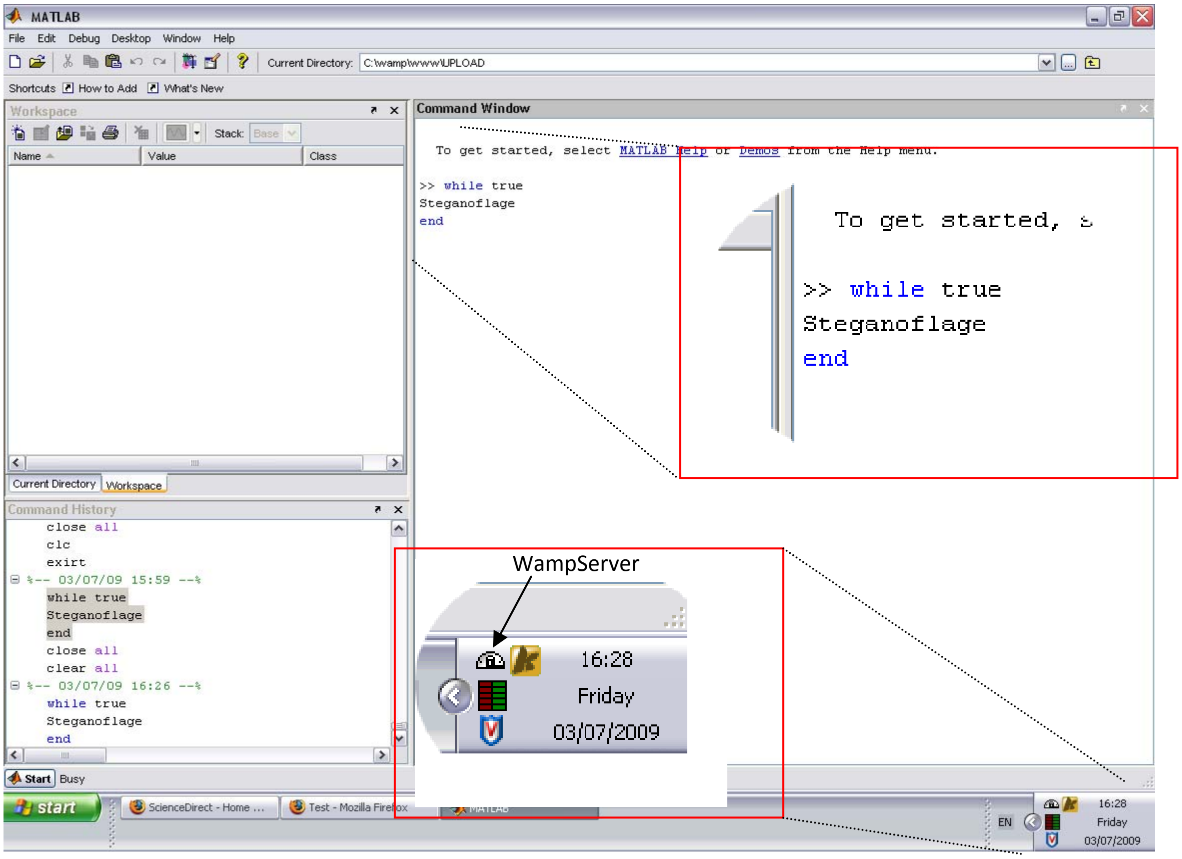
5.3 Bridging PHP to MATLAB

The MATLAB built-in function *MEX* compiles and links MATLAB source files into a shared library called a MEX-file. This file is executable from within MATLAB which takes advantage of the speed provided by C/C++. MATLAB does not have a toolbox which supports interface to web browsing languages like HTML, PHP or JavaScript.

This section aims to discuss the motivations behind this unfamiliar setup. The following points are of interest in this discussion:

- MATLAB functions are not complete, therefore instead of programming a new function completely, it is more efficient to utilize the functions available which are written in other languages. These can then be interfaced with the main function, e.g., SHA-2.php.
- Thousands of engineers and scientists using MATLAB want to put systems online to provide a 24 hour open platform to capture users input from around the world and allow online system interaction.
- For security reasons or intellectual property preservation, many MATLAB users would encourage a safe distribution of a demonstration of their completed system rather than the MATLAB source files. A PHP interface allows the end user to interact with the system without having to disclose core source files.
- Often the output to the browser is visually more pleasing in that it can include imaging special effects and 3D graphics. It is also more organised and helpful in generating a multimodal, e.g., text, images, video, and audio, encapsulation.

Figure 5.2 shows *Steganoflage* running within MATLAB where MATLAB acts as a server-side application. Shown in the figure is the WampServer which is a Windows web development environment, it allows creating web applications with Apache, PHP and the MySQL database (Bourdon, 2009). The “while” loop forces the function to continuously execute the command. A discussion will follow on how to parse standard HTML codes into MATLAB code which can then be extended to JavaScript, Java Applets and PHP tags. In this online version *Steganoflage* is set to consistently check for specific user input files. Specifically *Steganoflage* checks for a cover, a secret and a password. When these are found the underlying functions run automatically. At the end of the process, *Steganoflage* outputs the stego-image, copies the cover to another directory to display it to the browser and deletes all files which were input by the user. Figure 5.3 shows the online user-friendly interface of *Steganoflage* directly linked to the main function shown in Figure 5.2. After clicking “Encode” the next page shown in Figure 5.4 is displayed giving the user a clickable link to view the final results. Figure 5.5 shows the results view which is coded automatically using the subroutine shown in Figure A.1. The figure is showing the original image (right) and the stego-image (left).

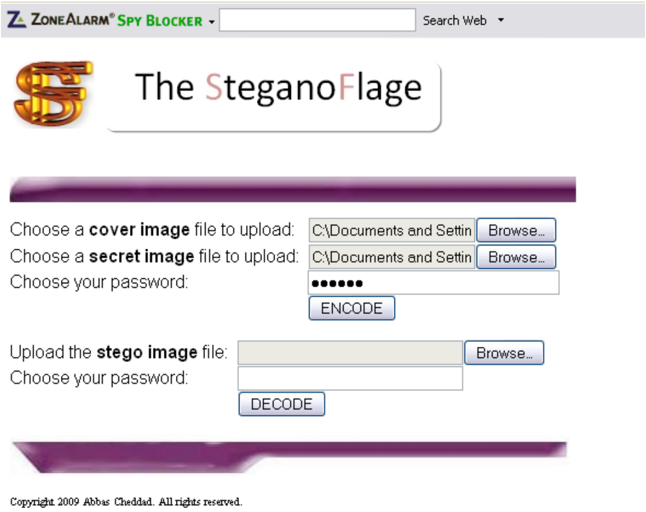


Current Directory: C:\wamp\www\UPLOAD

```
To get started, select MATLAB Help or Demos from the Help menu.

>> while true
Steganoflage
end
```

```
close all
clc
exirt
%-- 03/07/09 15:59 --%
while true
Steganoflage
end
close all
clear all
%-- 03/07/09 16:26 --%
while true
Steganoflage
end
```

Figure 5.2: *Steganoflage* running with WampServer running in the background



Choose a **cover image** file to upload: C:\Documents and Settin

Choose a **secret image** file to upload: C:\Documents and Settin

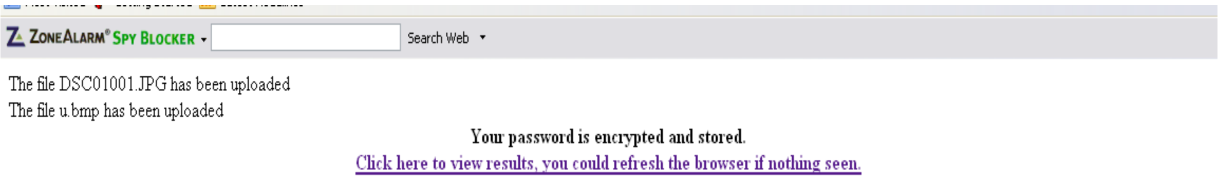
Choose your password: ●●●●●●

Upload the **stego image** file:

Choose your password:

Copyright 2009 Abbas Cheddad. All rights reserved.

Figure 5.3: *Steganoflage*'s online user interface

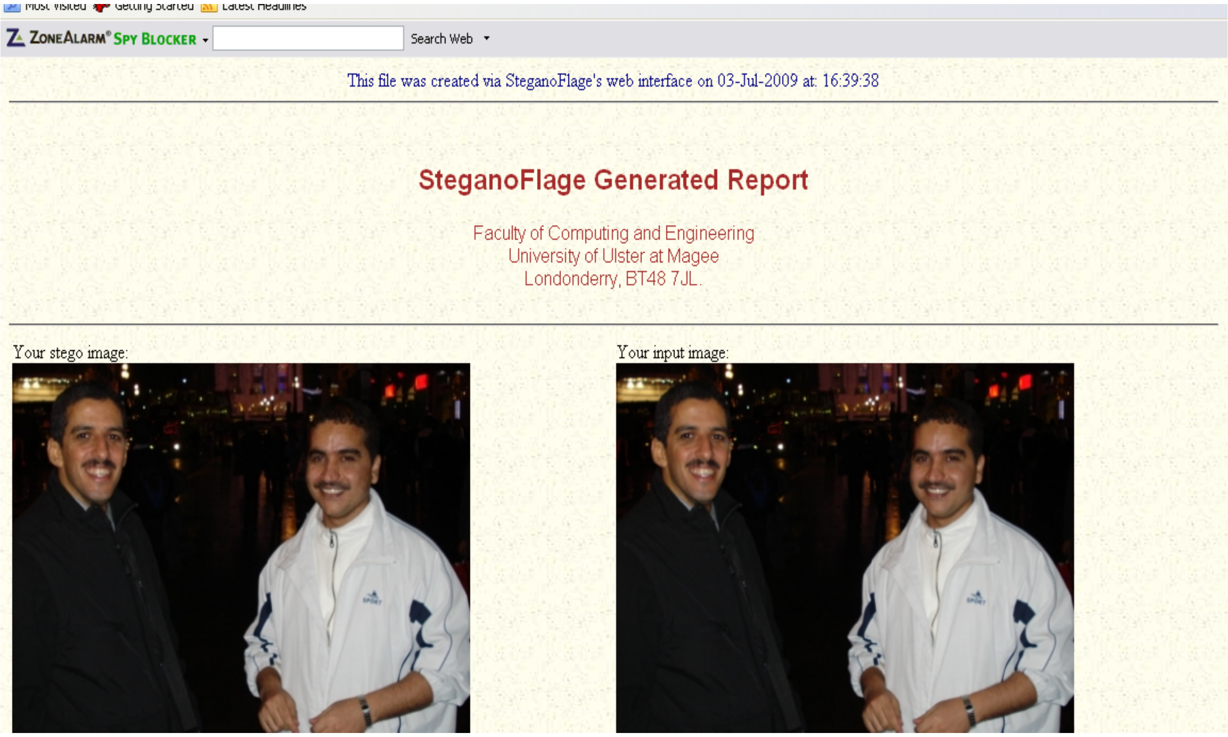


The file DSC01001.JPG has been uploaded
The file u.bmp has been uploaded

Your password is encrypted and stored.
Click here to view results, you could refresh the browser if nothing seen.

Figure 5.4: Hyperlink created to view results on the browser

In the offline application, the GUI was created in such a way that the user is prompted initially with the terms and conditions of utilizing the software, see Figure 5.6. Only when those conditions are accepted by the user does the main GUI pop-up, as shown in Figure 5.7.



This file was created via SteganoFlage's web interface on 03-Jul-2009 at: 16:39:38

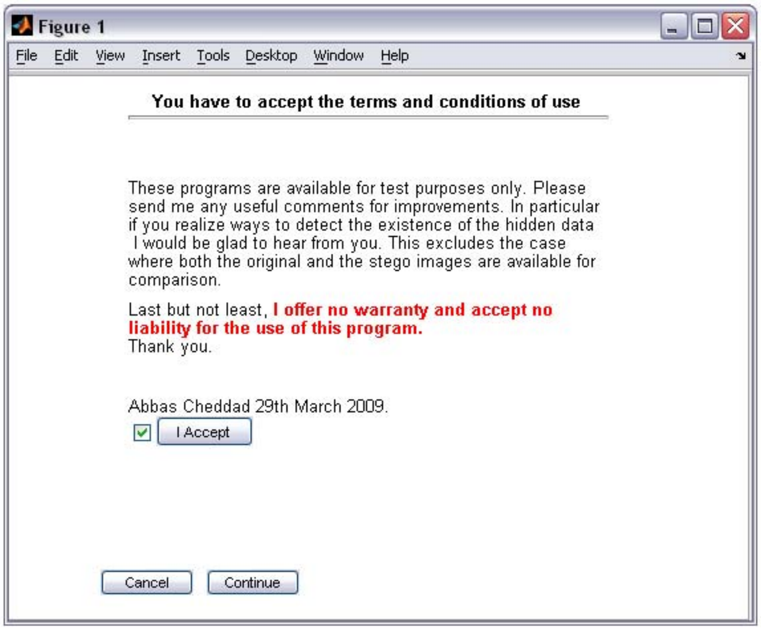
SteganoFlage Generated Report

Faculty of Computing and Engineering
University of Ulster at Magee
Londonderry, BT48 7JL.

Your stego image:

Your input image:

Figure 5.5: The generated results page “Report.html”



You have to accept the terms and conditions of use

These programs are available for test purposes only. Please send me any useful comments for improvements. In particular if you realize ways to detect the existence of the hidden data I would be glad to hear from you. This excludes the case where both the original and the stego images are available for comparison.

Last but not least, **I offer no warranty and accept no liability for the use of this program.**
Thank you.

Abbas Cheddad 29th March 2009.

Figure 5.6: User agreement

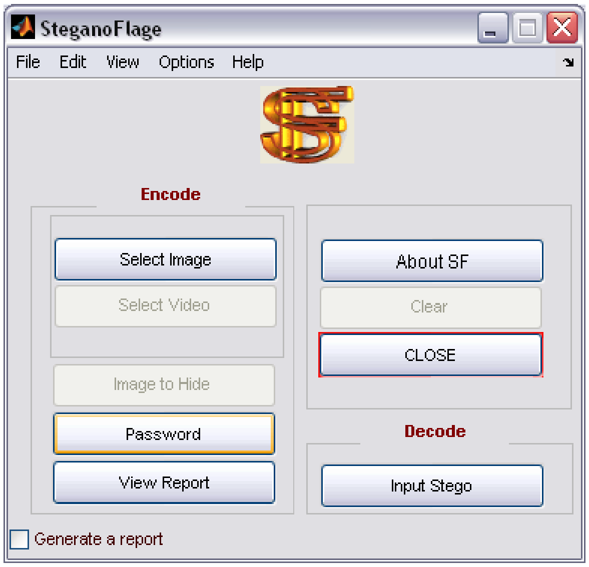


Figure 5.7: *Steganoflage*’s offline application

5.4 Applications of Steganoflage

A number of steganographic methods have been introduced, however, few authors have applied steganography and information hiding to real world problems with the exception of the works in (Ho & Shu, 2003), (Beşdok, 2005), (Zou et al., 2006) and (Lou et al., 2009). The objective in this section is to put into context practical applications of the research carried out on enhancing steganography in digital images that could solve some practical application problems. The following sub-sections discuss three potential applications, combating digital forgery, multilayer security for patients' data storage and transmission and finally digital reconstruction of lost signals. These applications take into account the steganographic enhancements discussed in Chapter 4 but without considering skin-tone areas.

5.4.1 Combating digital forgery

The recent digital revolution has facilitated communication, data portability and on-the-fly manipulation. Unfortunately, this has brought with it some critical security vulnerabilities that put digital imagery at risk. "While we may have historically had confidence in the integrity of this imagery, today's digital technology has begun to erode this trust" (Farid, 2009). The problem is in the security mechanism adopted to secure these documents by means of encrypted passwords. This security shield does not actually protect the documents which are stored intact. Historically, the forgery of a document was done mechanically, however, since the recent boost in communication technology, the massive increase in database storage and the introduction of e-Government, documents are increasingly being stored in a digital form. This goes hand in hand with the aim of the paperless workspace, but it does come at the expense of security breaches especially if the document is transmitted over a network. Document forgery is a worry for a range of organisations such as governments, universities, hospitals and banks (Cheddad et al., 2009b). The ease of digital document reproduction and manipulation has attracted many eavesdroppers.

Motivations

Relational Database Management Systems, RDBMS, secure scanned documents through the use of a password linked to the database. This means that scanned documents are stored with a 'string' encrypted password. The main problem arises if a hacker is able to crack the password and is then able to modify any document digitally and log out as if nothing has happened. In July 2005 it was discovered that a number of Second World War files held at the National Archives contained forged documents. An internal investigation found that the forgery took place during or after the year 2000 (National-Archive, 2008). The method developed in this thesis “Steganflage” could be used to help solve this real world problem through the special case of steganography “self-embedding”.

Information hiding is used for owner identification, royalty payments, and authentication by determining whether the data has been altered in any manner from its original form (Zhao et al., 2003). Popescu (Popescu, 2005) shows a comprehensive investigation carried out on image forensics which aims to detect forgery by means of the preserved natural image statistics. Although, the authors seem to have successfully created systems, such as (Shefali et al., 2008), whereby image forgery can be detected the *Steganoflage* method also shows what the original ‘non-forged’ image looked like. In some cases, for instance in court, it is not sufficient to just be able to tell that the image or document has been tampered with, which can be caused by a legitimate process such as JPEG compression, without giving the jury a tool to actually extract the original document.

Lukáš (Lukáš et al., 2006) take another approach to detecting forgery through the presence of the camera pattern noise, which is a unique stochastic characteristic of imaging sensors, in individual regions in the image. The forged region is determined as the one that lacks the pattern noise. The authors assume the availability of either the same camera that took the attacked image or another image taken with the same camera. The method deals with the detection without the recovery and suffers from false alarms. As far as image forgery is concerned this approach has no practical soundness as it cannot be generalised.

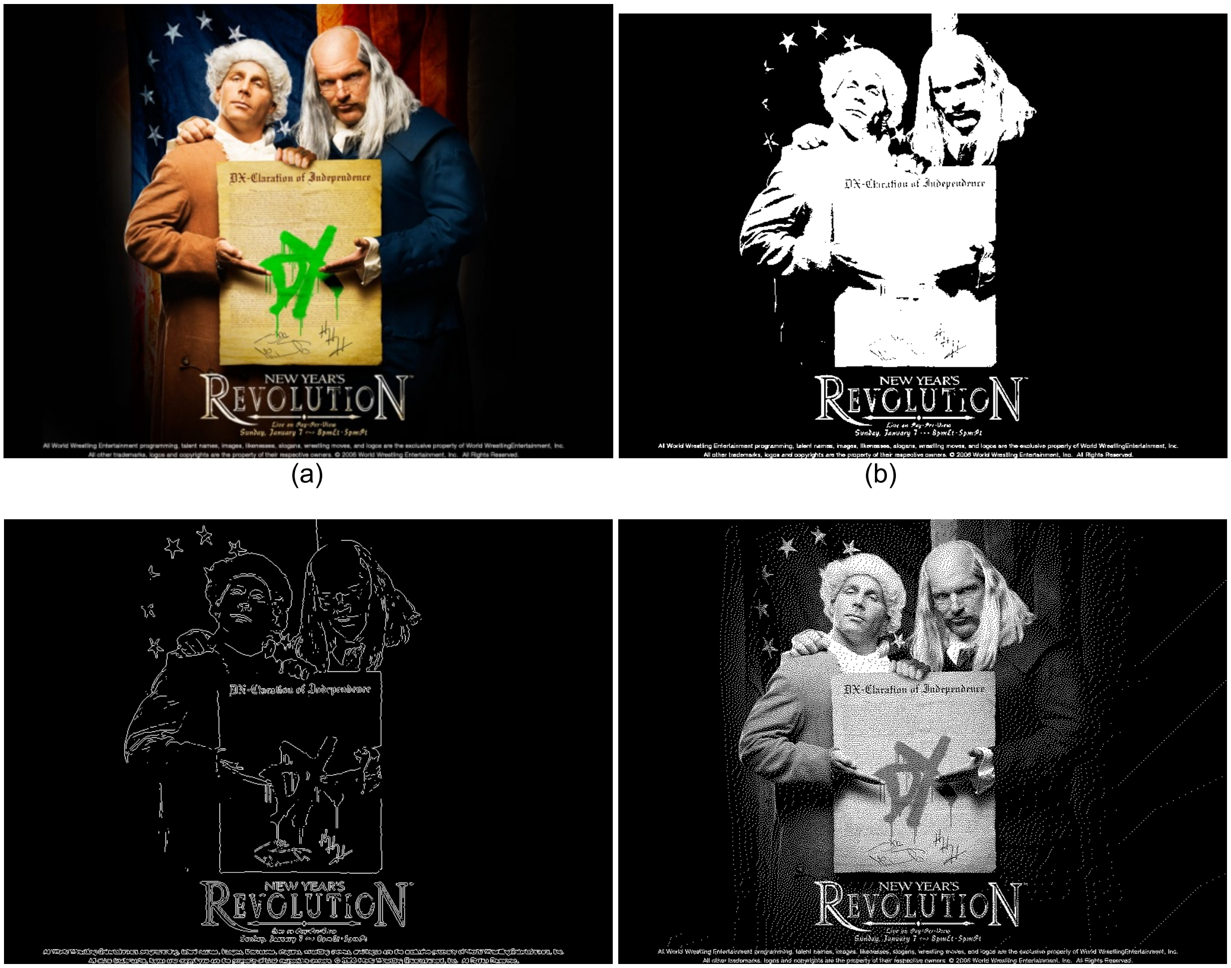
Most of the preceding algorithms deal with image authentication and pay little attention to recovery, for example the work in (Deguillaume et al., 2003) and (Fridrich et al., 2003). Those which address recovery use a block-wise-based recovery process. The block-based recovery is based on the assumption that the forged segment will likely be a connected component rather than a collection of very small patches or individual pixels. Examples of block-based algorithms are (Fridrich & Goljan, 1999),(Fridrich & Goljan, 1999b) and (Fridrich et al., 2003).

Methodology

Since means are needed to protect scanned documents against forgery it is essential that the payload will carry as much information from the host, cover image, as possible. There is a trade-off between perceptual visualization and space demand for embedding, usually measured in bits. Without taking compression into account, the payload can be consistent with the cover signal, therefore, if the cover is stored as an 8-bit unsigned integer type then the payload will require 8 templates when applying the one bit substitution method. There is a high payload steganography approach called ABCDE (Hioki, 2002), but it is prone to statistical attacks as it acts in the spatial domain. Moreover it cannot resist any kind of manipulation to the stego-image.

An approximation of the cover document can be achieved by applying the gray threshold technique which results in a binary image demanding only one bit per pixel for storage. Some authors suggest using an edged image instead as it approximates the cover better. In the search for the best way to represent the cover image with the least bit requirement for embedding the technique of dithering was identified as an ultimate pre-processing step which is the foremost task in building *Steganoflage*. The process can be regarded as a distorted quantization of colours to the lowest bit rate. Meanwhile, reduction of the number of image colours is an important task for transmission, segmentation, and lossy compression of colour visual information (Li et al., 2003) which is why dithering is used for printing.

Dithering is a process by which a digital image with a finite number of gray levels is made to appear as a continuous-tone image (Farid, 2008). For instance Figure 5.8 (a) shows a 24-bit image, RGB image, and its three binary representations. Even though in all these binary maps each pixel takes on only one bit, it is apparent that the way dithering quantizes image pixels contributes considerably to the final quality of data approximation. It is observed that thresholding performs better in text based documents, while in capturing graphics it is proven to be a poor performer compared to dithering. Dithering is utilised since the aim is to produce a general workable prototype within which the presence of both text and graphics must be considered.



(a) (b)

(c) (d)

Figure 5.8: Image fidelity in different binary representations. (b, c and d) are three one bit images, i.e., binary, of the greyscale version of *(a)*. (b) is created by thresholding, (c) is created by an edge operator, and (d) is created by dithering respectively

There exist different algorithms to generate an inverse halftone image, among which are: look-up table based methods (Mese & Vaidyanathan, 2001), filtering-based methods and projection-based methods (Luo et al., 2008). As an improved version of the filtering based methods, Neelamani (Neelamani et al., 2000) and (Neelamani et al., 2009) propose an inverse halftoning in the Wavelets domain. All of the above make use of Floyd-Steinberg (Floyd & Steinberg, 1975) and Jarvis (Jarvis et al., 1976) kernels to generate the error diffusion signal. A test was carried out to select which algorithm performs better. Based on Table 5.1 it can be seen that the Jarvis implementation in the Wavelets domain provides better performance.

Table 5.1: Performance of different inverse halftoning algorithms

Algorithm	Performance on Lena image measured using the PSNR (dB)
Floyd: Classic raster scan	28.084
Floyd: Raster scan with edge enhancement	28.2513
Floyd: Serpent scan	27.6623
Jarvis: Serpent scan	26.6602
Jarvis: Raster scan	27.221
Jarvis: Wavelets	28.292

Figure 5.9 shows an example of self-embedding. Figure 5.10 shows another example applied to a scanned document showing the same process. Additional examples are shown in Figure C.1, Figure C.2, Figure C.3 and Figure C.4 in the Appendices. The proposed embedding scheme is robust against reasonable noise load that can be introduced for example during electronic transmission of the stego-document. Moreover, using DWT gives the advantage of being able to convert the stego-document into lossy compressed formats such as JPEG, without having to lose so much detail.

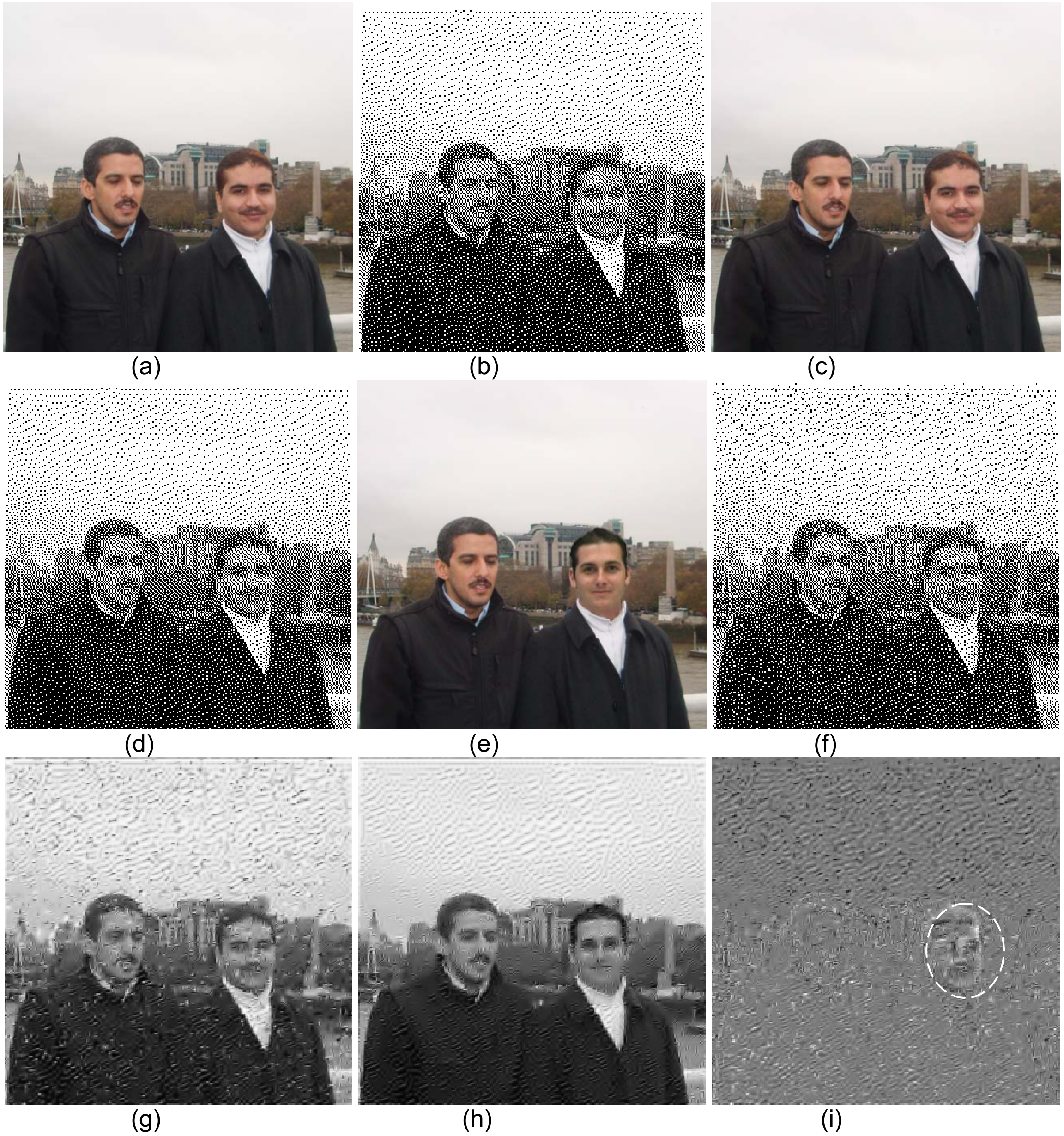
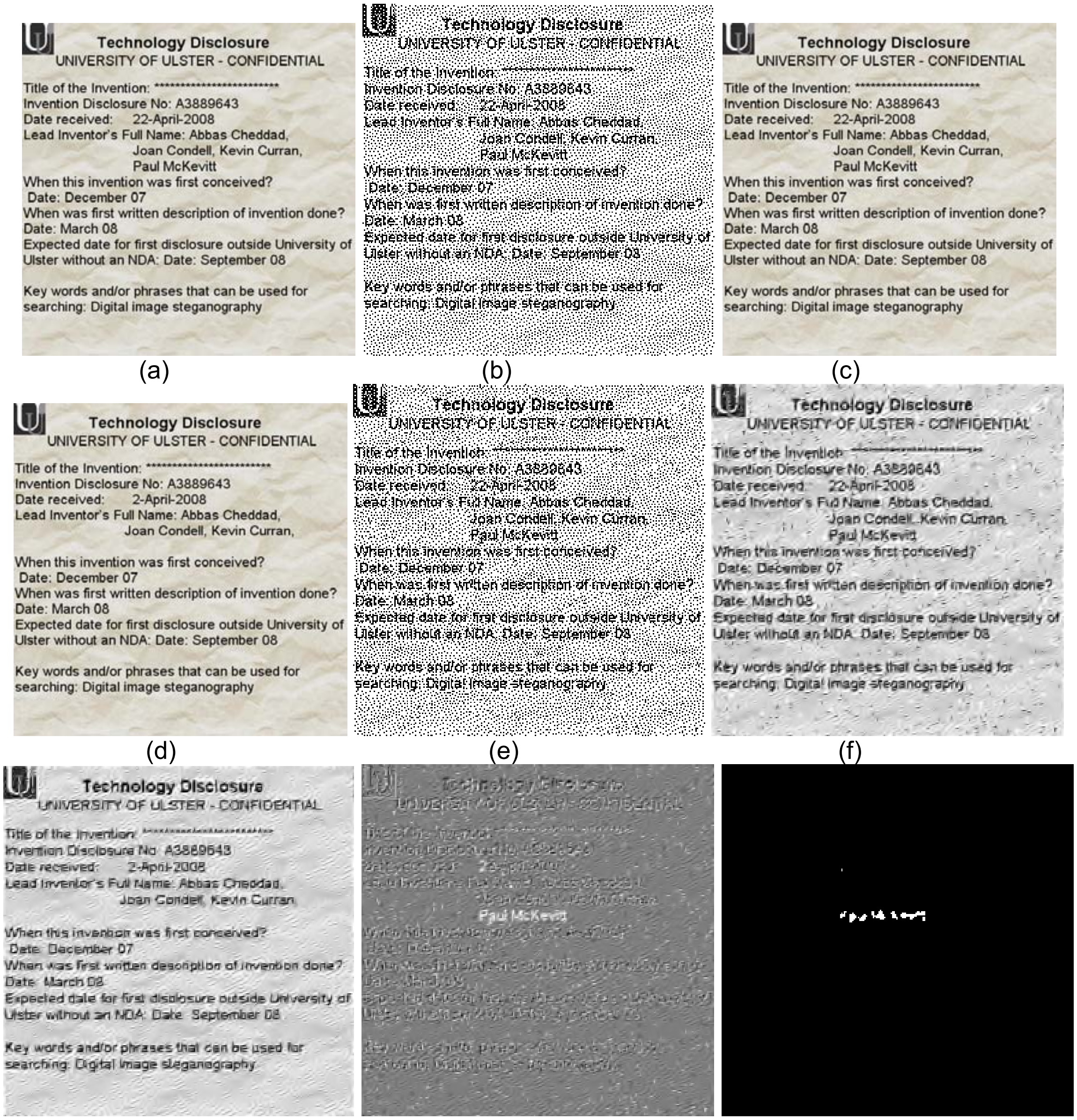


Figure 5.9: Self-embedding, Example 1. (c) shows a perceptually identical copy of *(a)* with a duplicate of itself embedded into its pixels, (d) confirms that the embedded data can be retrieved intact, (e) simulates image tampering where the face of the person in *(c)* has been altered digitally, (f) is the extracted copy from *(e)*, (i) is the error signal derived from subtracting *(g)* and *(h)* which are the inverse dithered versions of the extracted copy and the direct half-tone of *(e)*, respectively. Notice that only the tampered region, herein shown within a superimposed circle, demonstrates a coherent object in (i).

Table 5.2 shows the proposed algorithm outperforming other related methods pertaining to visual distortion of the carrier image shown in Figure 5.11. Figure 5.11 depicts: (left) original image Lena (512x512) and (right) a self-embedded Lena, PSNR=41.9480 dB.



Technology Disclosure
UNIVERSITY OF ULSTER - CONFIDENTIAL
Title of the Invention: ************************
Invention Disclosure No: A3889643
Date received: 22-April-2008
Lead Inventor's Full Name: Abbas Cheddad,
Joan Condell, Kevin Curran,
Paul McKevitt
When this invention was first conceived?
Date: December 07
When was first written description of invention done?
Date: March 08
Expected date for first disclosure outside University of Ulster without an NDA: Date: September 08

Key words and/or phrases that can be used for searching: Digital image steganography

(a)

Technology Disclosure
UNIVERSITY OF ULSTER - CONFIDENTIAL
Title of the Invention: ************************
Invention Disclosure No: A3889643
Date received: 22-April-2008
Lead Inventor's Full Name: Abbas Cheddad,
Joan Condell, Kevin Curran,
Paul McKevitt
When this invention was first conceived?
Date: December 07
When was first written description of invention done?
Date: March 08
Expected date for first disclosure outside University of Ulster without an NDA: Date: September 08

Key words and/or phrases that can be used for searching: Digital image steganography

(b)

Technology Disclosure
UNIVERSITY OF ULSTER - CONFIDENTIAL
Title of the Invention: ************************
Invention Disclosure No: A3889643
Date received: 22-April-2008
Lead Inventor's Full Name: Abbas Cheddad,
Joan Condell, Kevin Curran,
Paul McKevitt
When this invention was first conceived?
Date: December 07
When was first written description of invention done?
Date: March 08
Expected date for first disclosure outside University of Ulster without an NDA: Date: September 08

Key words and/or phrases that can be used for searching: Digital image steganography

(c)

Technology Disclosure
UNIVERSITY OF ULSTER - CONFIDENTIAL
Title of the Invention: ************************
Invention Disclosure No: A3889643
Date received: 2-April-2008
Lead Inventor's Full Name: Abbas Cheddad,
Joan Condell, Kevin Curran,

When this invention was first conceived?
Date: December 07
When was first written description of invention done?
Date: March 08
Expected date for first disclosure outside University of Ulster without an NDA: Date: September 08

Key words and/or phrases that can be used for searching: Digital image steganography

(d)

Technology Disclosure
UNIVERSITY OF ULSTER - CONFIDENTIAL
Title of the Invention: ************************
Invention Disclosure No: A3889643
Date received: 22-April-2008
Lead Inventor's Full Name: Abbas Cheddad,
Joan Condell, Kevin Curran,
Paul McKevitt
When this invention was first conceived?
Date: December 07
When was first written description of invention done?
Date: March 08
Expected date for first disclosure outside University of Ulster without an NDA: Date: September 08

Key words and/or phrases that can be used for searching: Digital image steganography

(e)

Technology Disclosure
UNIVERSITY OF ULSTER - CONFIDENTIAL
Title of the Invention: ************************
Invention Disclosure No: A3889643
Date received: 22-April-2008
Lead Inventor's Full Name: Abbas Cheddad,
Joan Condell, Kevin Curran,
Paul McKevitt
When this invention was first conceived?
Date: December 07
When was first written description of invention done?
Date: March 08
Expected date for first disclosure outside University of Ulster without an NDA: Date: September 08

Key words and/or phrases that can be used for searching: Digital image steganography

(f)

Technology Disclosure
UNIVERSITY OF ULSTER - CONFIDENTIAL
Title of the Invention: ************************
Invention Disclosure No: A3889643
Date received: 2-April-2008
Lead Inventor's Full Name: Abbas Cheddad,
Joan Condell, Kevin Curran,

When this invention was first conceived?
Date: December 07
When was first written description of invention done?
Date: March 08
Expected date for first disclosure outside University of Ulster without an NDA: Date: September 08

Key words and/or phrases that can be used for searching: Digital image steganography

(g)

Paul McKevitt

(h)

(i)

Figure 5.10: Self-embedding, Example 2. (c) shows a perceptually identical copy of *(a)* with a duplicate of itself embedded into its pixels, (d) simulates image tampering where the date and inventor name, Paul Mc Kevitt, in *(c)* have been forged digitally, (e) is the extracted copy from *(d)*, (h) is the error signal derived from subtracting *(f)* and *(g)* which are the inverse dithered versions of the extracted copy and the direct half-tone of *(d)*, respectively, (i) is a global threshold applied to *(h)*

Table 5.2: Visual distortion of the cover using proposed algorithm and other methods

	PSNR (dB)			
Image	Proposed	(Shao et al., 2001)	(Kostopoulos et al., 2002)	(Lin & Chang, 2001)
Lena	41.9480	34.35	35.10	38.0164*

* We selected a balance between robustness and visual distortion in the tool's setting.

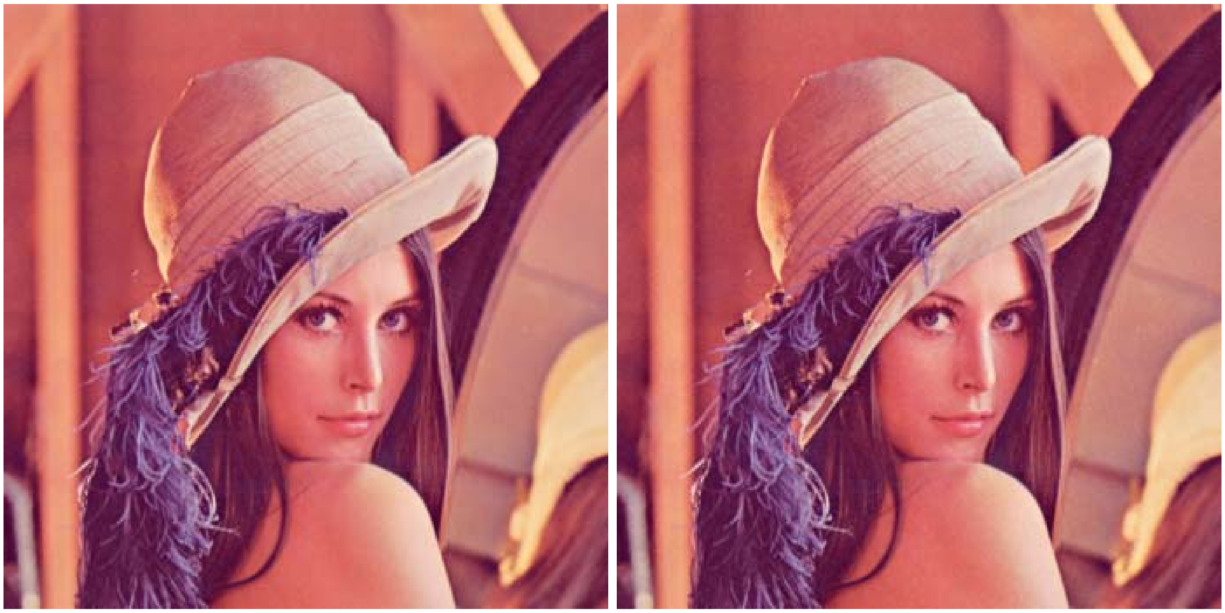


Figure 5.11: Visual distortion of Lena image. (Left) original image and (right) stego-image

A general graphical scheme showing the benefit that may arise from adopting the proposed method is shown in Figure 5.12.

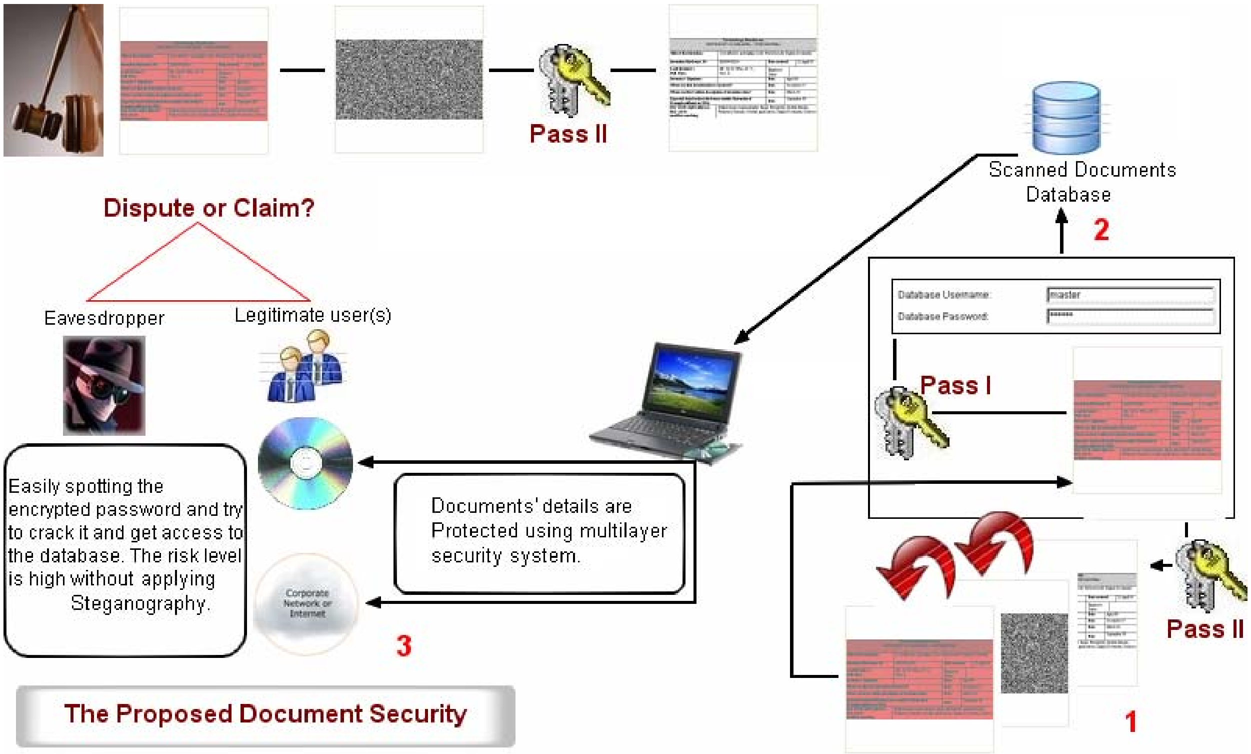

Pass II
Dispute or Claim?
Eavesdropper
Legitimate user(s)
Scanned Documents Database
2
Database Username: master
Database Password: ******
Pass I
Easily spotting the encrypted password and try to crack it and get access to the database. The risk level is high without applying Steganography.
Documents' details are Protected using multilayer security system.
Corporate Network or Internet
3
The Proposed Document Security
Pass II
1

Figure 5.12: The advantage of the proposed algorithm for securing scanned document

5.4.2 Multilayer security for patients’ data storage and transmission

Electronic patient records, EPRs, are some of the most precious entities in a health care centre. Steganography is an enabling technology that can assist in transmitting EPRs across distances to hospitals and countries through the Internet without worrying about security breaches on the network, such as eavesdroppers’ interception. Thus, embedding the patient’s information in the image could be a useful safety measure. Medical records of patients hold exceptionally sensitive information that requires a rigid security during both storage and transmission. The stego-image carrying the patient data as shown in Figure 5.13 would not draw attention if transmitted. Figure 5.13 shows: (a) a CT scan image of a patient with patient’s information, (b) encrypted secret data (payload) of (a), a clean image showing nature in which the encrypted data will be embedded to and finally (d) shows the stego-image carrying the encrypted patient data.

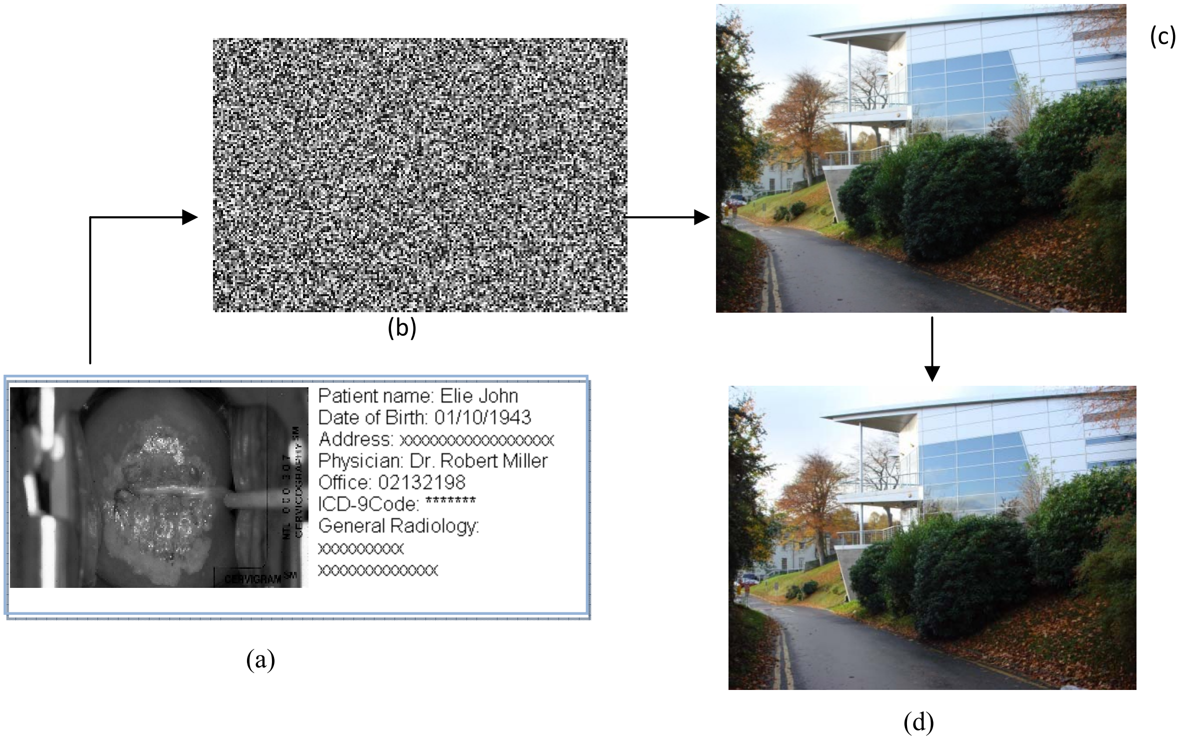


Figure 5.13: Embedding EPRs (Electronic Patient Records) data in innocuous looking image

5.4.3 Digital reconstruction of lost signals

Current research aimed at repairing audio streams relies on improving ‘Quality of Service’ protocols or masking gaps in the stream with linear interpolation techniques (Doherty, 2009). The encryption and the hiding strategy of *Steganoflage* can be tailored to act as an intelligent streaming audio/video system that uses techniques to conceal transmission faults from the listener that are due to lost or delayed packets on wireless networks with bursty arrivals, providing a disruption tolerant broadcasting channel. The following exemplifies a theoretical model, the *chirp* audio signal which comes with the MATLAB package is shown in Figure 5.14. The core idea here is to divide the signal into static ranges, each of which is compressed and embedded into its preceding part. If any part does not play, its hidden version will be retrieved, even with no transmission, from the stored corresponding part.

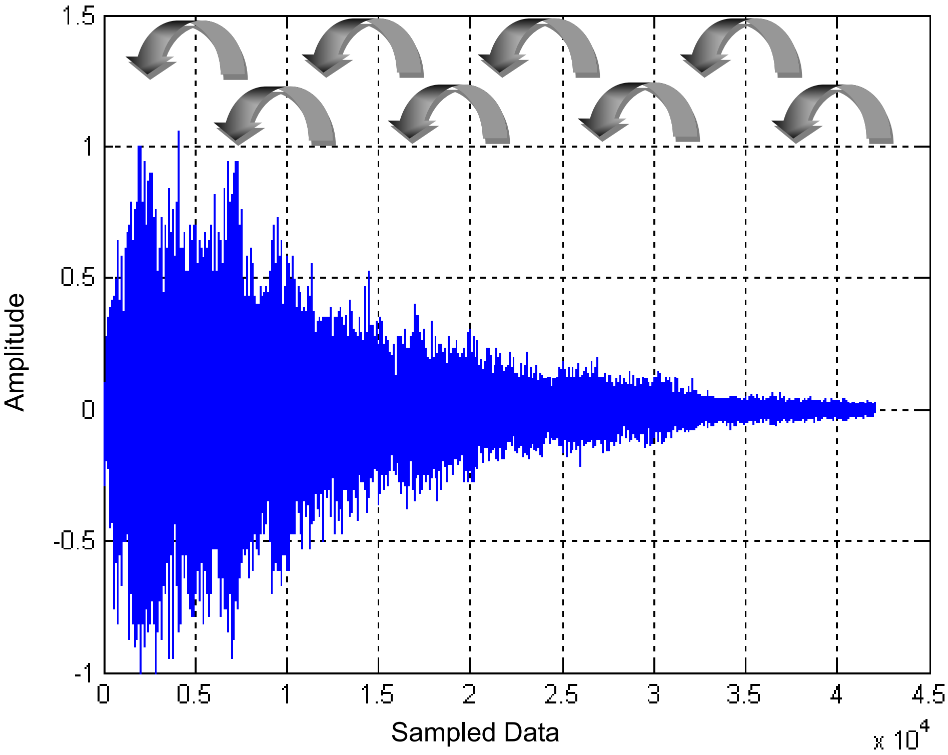


Figure 5.14: Audio error concealment model using information hiding. The core idea here is to divide the signal into static ranges, each of which is compressed and embedded into its preceding part

The main concern here is to allow for a quasi-reconstruction of a lost signal. For the human auditory system an uninterrupted sound is necessary. Therefore, an encrypted lossy compressed signal can be embedded in each portion recursively. In the following example, for simplicity and to comply with the JPEG file format, only data with positive amplitude is considered. Figure 5.15 visualizes a compressed audio signal in JPEG format with compression quality factor $Q = 55\%$.

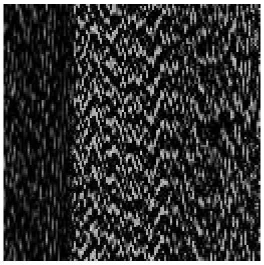
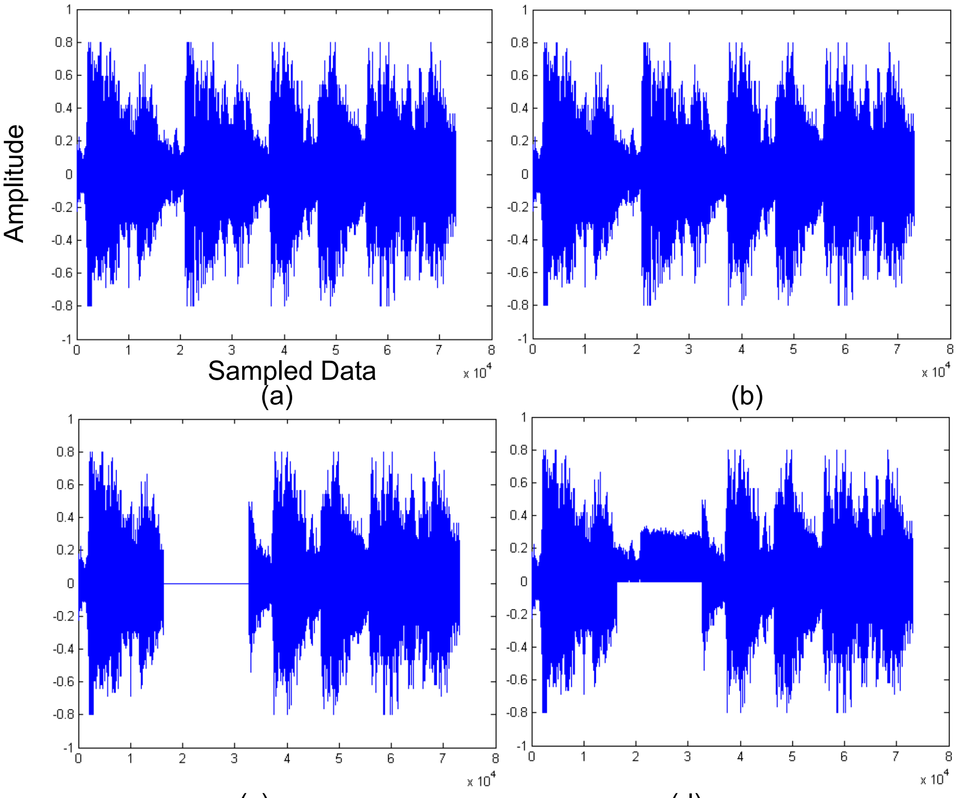


Figure 5.15: JPEG compressed visualization of the audio signal

This audio signal is used to recover the lost portion, see Figure 5.16 (d), which comprises 22% of the total length of the audio track. Next, the compressed audio is further encrypted using the method highlighted in Chapter 4. The encrypted bit stream is embedded into the original audio to produce a stego-audio as shown in Figure 5.16 (b). Figure 5.16 shows: (a) original audio signal, (b) stego-audio carrying an encrypted and lossy compressed copy of 22.4091% of the total audio length, (c) dropped signal and (d) recovered approximation of the lost signal. Amplitude values are generally in the range [-1, +1]. A simulation of a lost signal is achieved by setting all values in a certain range to '0' resulting in a silence period for some seconds as shown in Figure 5.16 (c). Figure 5.16 (d) illustrates the recovery of the embedded amplitude. Even though the recovered signal has deformed the original due to JPEG compression, Q=55%, and has discarded all the data with negative amplitude values, the playback assures continuity without disruption. The performance can be enhanced further using error-diffusion techniques, half-toning, instead of JPEG compression.



(a) (b)

(c) (d)

Figure 5.16: Experiment on audio signal quasi-recovery. (a) original audio signal, (b) stego-signal, (c) dropped signal and (d) recovered signal

In the same way video streaming can also benefit from such a model. Embedding halftone colour frames into another group would suffice as the inverse-halftoning could approximate the original 'lost' frames with a very high degree of precision. The viewer might not even notice the replaced frames. This is a very exciting research area that needs further development. Other methods have been used to deal with this problem like the system of the WCAM, Wireless Cameras and Seamless Audiovisual Networking, project shown in Figure 5.17. The figure shows: (top) WCAM project at the University of Bristol (Signal Processing Group, 2008) and (bottom) Spatio-temporal error affecting YouTube video streaming when switched to high definition.

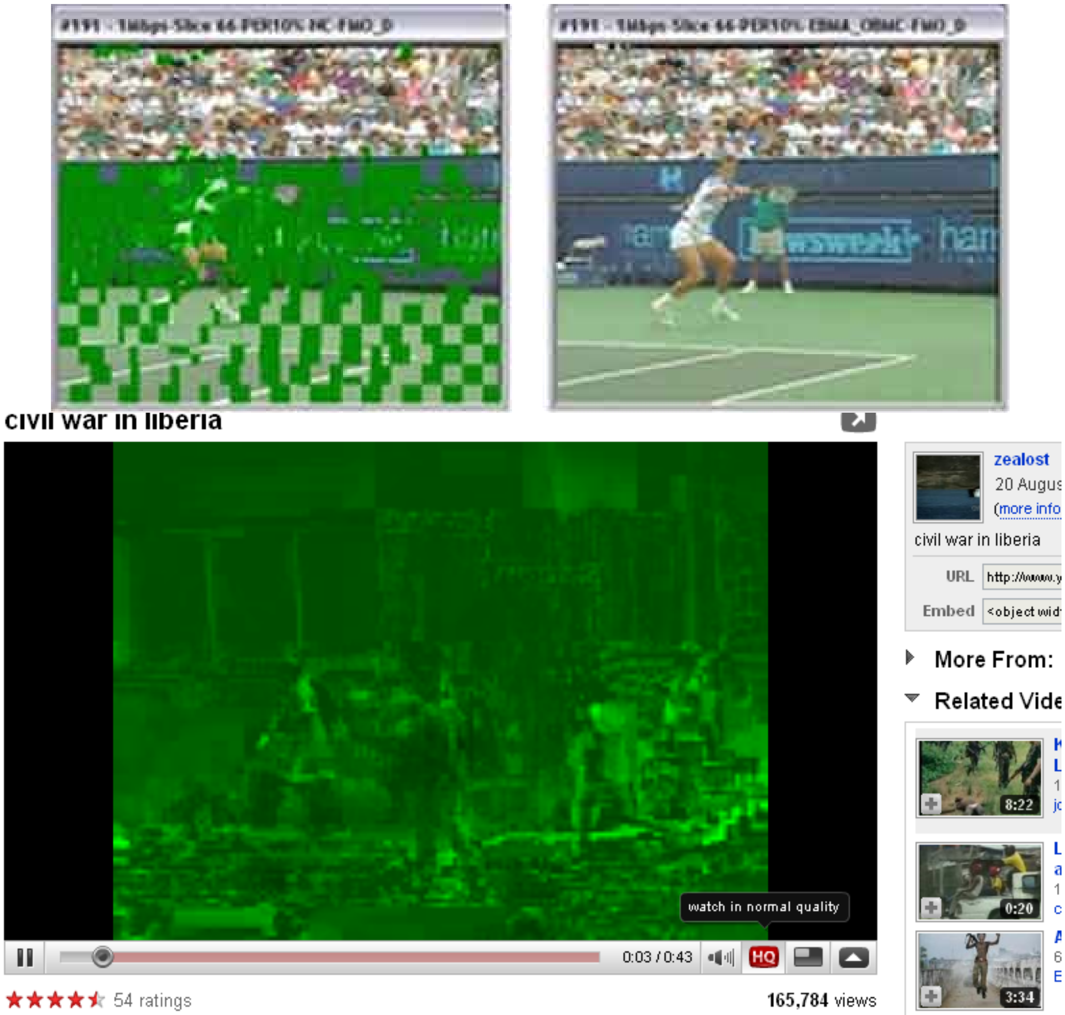


Figure 5.17: Error concealment in video streaming. (Top-left) video with error during transmission, (top-right) recovered blocks and (bottom) video streaming errors in YouTube

5.5 Summary

This chapter discusses the system design and architecture of *Steganoflage* and the way in which it bridges MATLAB to web scripting languages. It also discusses applications to real world problems. The first application outlines counterfeiting deterrence of sensitive or secure documents of value, an approach to scanned document forgery detection and a correction method which uses an information hiding technique that is highly secure, efficient and robust to various image processing attacks. It is a novel approach to allow documents to repair themselves after a forgery attack. The payload, which is a dithered version of the cover, has a low bit rate while capturing the main image characteristics needed for reconstruction. The second application involves patients' medical data where *Steganoflage* offers a tightened security mechanism. Error concealment in audio and video streaming technology and wireless broadcasting are also discussed as a possible application of *Steganoflage*. Audio steganography is not the focus of this thesis but Figure 5.15 shows that an audio signal can be treated as an image on which *Steganoflage* can act.

Chapter 6 unfolds the robustness of Steganoflage and proves that through experimental results derived from universal testing methods.

CHAPTER

SIX

Experimental Results

This chapter discusses the evaluation of the *Steganoflage* system. *Steganoflage* involves different components which necessitate the division of this chapter into three main sections. First, the security of the proposed image encryption algorithm is analysed and compared to existing methods. Then the performance of the skin-tone detection method is explored. Finally an overall evaluation of the full proposed *Steganoflage* system is detailed.

6.1 Security Analysis of the Image Encryption Method

This section analyses the security aspects of the proposed method. Encryption algorithms are assumed to be robust to different statistical and visual attacks. Moreover the key sensitivity and key space should be adequate. In addition, being a tailored method for steganography, the result should exhibit high randomness and balanced bit values. This section is split into six sub-sections, namely, key space analysis, key sensitivity analysis, adjacent pixels analysis, the randomness test, differential analysis, and other security issues.

6.1.1 Key space analysis

The key space analysis of the proposed algorithm essentially involves analysing the underlying SHA-2 algorithm. SHA-2 accepts any key of any length less than 264 bits. SHA is secure because it is computationally infeasible to find a message which corresponds to a given message digest, or to find two different messages which produce the same message digest (SHA, 2001). SHA has been extensively adopted in several organisations and has received much scrutiny from the cryptography community. The proposed encryption algorithm is flexible enough to migrate to a newer version of the SHA's algorithmic family or other secure hash functions as it is known that no collisions have been found in SHA-2.

6.1.2 Key sensitivity analysis (malleability attack)

As planned in the design stage, the algorithm is proven to be very sensitive to initial conditions. This test is done by slightly modifying the key to decrypt data and see if the output shows any visual correlations with the one with the correct key. This test is shown in Figure 6.1. This immunity to malleability attack is due to the use of SHA and the FFT algorithms. See Figure D.1, Figure D.2 and Figure D.3 in Appendix D for additional examples to confirm this.

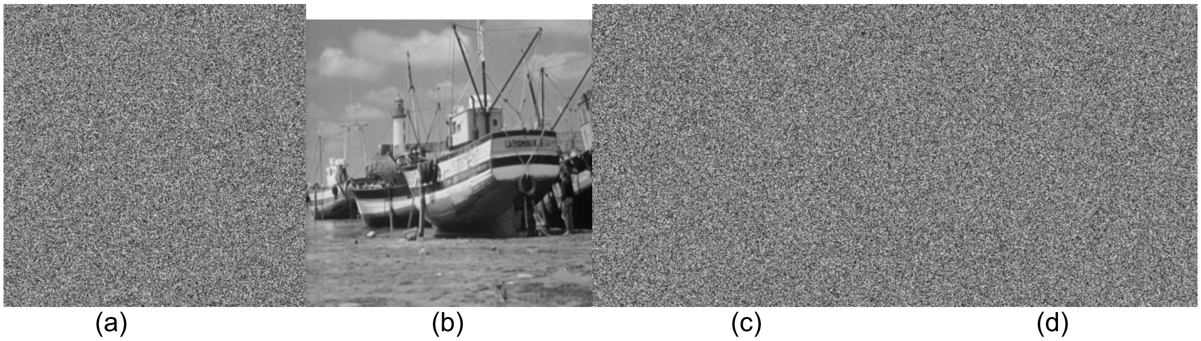

(a) (b) (c) (d)

Figure 6.1: Key sensitivity test. (a) the encrypted image; (b) the decrypted image *(a)* using the key 'Steganography' '40662a5f1e7349123c4012d827be8688d9fe013b'; (c) the decrypted image (a) using the wrong key 'Steganographie' 'c703bbc5b91736d8daa72fd5d620536d0dfbfe01'; (d) the decrypted image *(a)* using a slightly modified hash '40662a5f1e7349123c4012d827be8688d9fe013B'

6.1.3 Adjacent pixels analysis

To test the statistical properties of the original image and the encrypted version tests were carried out based on the linear relationship between two adjacent pixels horizontally, vertically and diagonally. It is observed that natural images with natural data have a high correlation ratio between neighbouring pixels. Figure 6.2 depicts a correlation analysis of 5000 pairs of horizontal adjacent pixels chosen randomly from:

(right) the original plain boat image, Figure 6.1 (b), showing that the image has high correlation between adjacent pixels which is of no surprise in natural images (left) the encrypted image, Figure 6.1 (a), using the proposed method, the correlation is very weak in the encrypted image, this is seen in the scattered plot of intensity values.

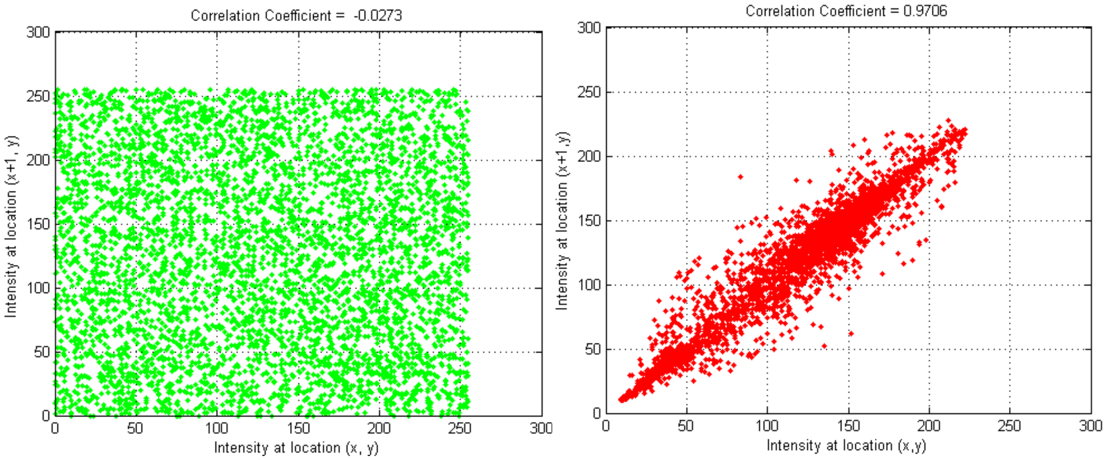


Figure 6.2: Correlation analysis of 5000 pairs of horizontal adjacent pixels

To measure this relationship, the correlation coefficient was calculated of each pair of pixels, as shown in Table 6.1.

Table 6.1: Comparison of correlation analysis with recent methods using Lena image

Method/Scan Direction	*Horizontal*	*Vertical*	*Diagonal*
Original Image	0.9194	0.9576	0.9016
PRNG	0.002291	0.005702	0.007064
(Wong et al., 2008b)	0.002933	-0.004052	0.001368
(Wong et al., 2008a)	0.006816	0.007827	0.003233
(Lian et al., 2005)	0.005343	0.008460	0.003557
(Zou et al., 2005)	0.01183	0.00872	0.01527
(Zeghid et al., 2006)	0.02	0.03	Not reported
(Wang & Zhang, 2008)	0.0085	0.0054	0.0242
(Tong et al., 2009)	0.0171	0.0098	0.0330
(Mazloom & Eftekhari-Moghadam, 2009)	0.01183	0.00016	0.01480
Proposed	-0.0028	-0.0068	0.0044

The comparison given in Table 6.1 shows that the proposed method outperforms other recent methods reported in the literature. Correlation coefficients, ranging from ‘1’ highly correlated to '-1' highly uncorrelated or a negative image, of pairs of adjacent pixels in different directions. These coefficients ensure the two considered images are statistically independent. To establish a fair evaluation the same test image was used. In the horizontal, diagonal and vertical directions the encrypted version under this scheme had the highest performance. Bear in mind that unlike various methods, the proposed algorithm does not involve an extensive and computationally intensive iterative process. The encrypted image shown in Figure 6.1 is automatically generated once the program is invoked with a key. The process does not retain any image statistics of the original image. This can be seen by comparing histograms of the plain and encrypted images, the original histogram is flattened and has a uniform distribution as shown in Figure 6.3. The figure shows: (top) natural image and its histogram and (bottom) the same image encrypted and the generated histogram.

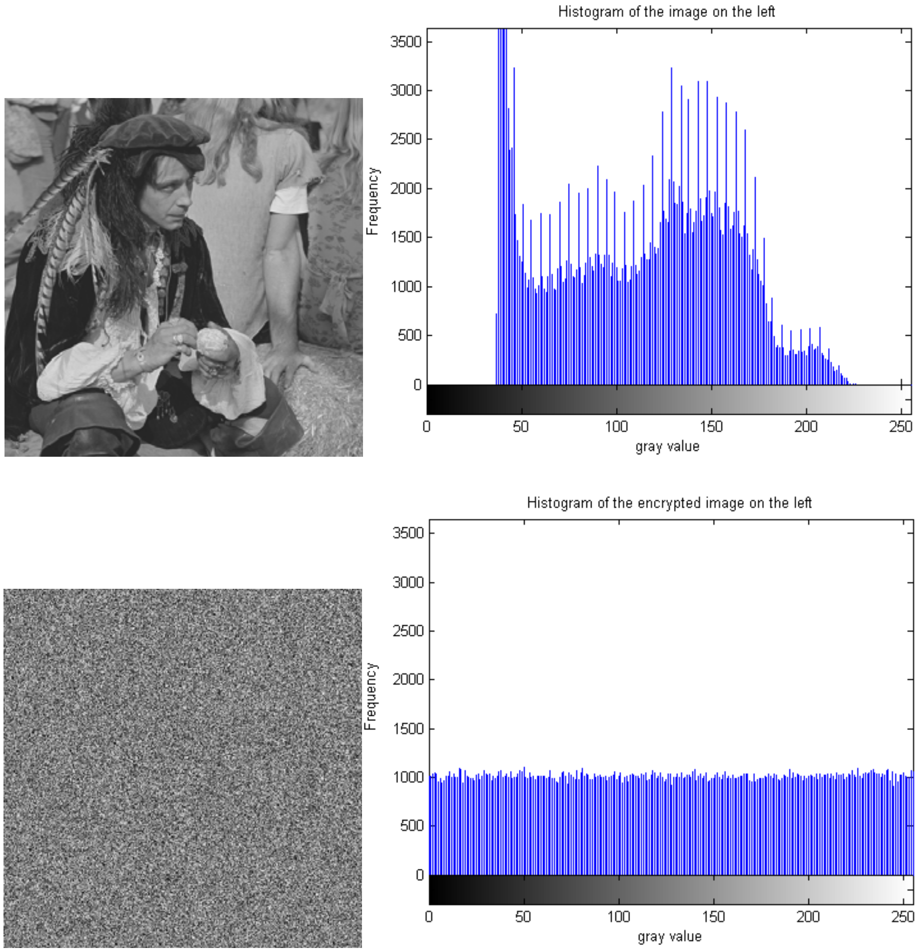


Figure 6.3: Eradication of image statistics

The paramount property of the proposed encryption algorithm in terms of adjacent pixels analysis is deemed important since, according to Lian (Lian et al., 2005), the high correlation information between adjacent pixels is the key behind the good performance of known/chosen plaintext attacks.

6.1.4 Randomness test / Distinguishing attack

In the randomness test the method is submitted to a range of empirical tests which measure the quality of the generated random sequence. “It is impossible to give a mathematical proof that a generator is indeed a random bit generator” (Menezes et al., 1996). There are many possible statistical tests which could be carried out, each of which reports the presence or absence of a “pattern” which, if detected, would indicate that the sequence is non random (Rukhin et al., 2008). Therefore, “the security of a stream cipher is closely connected to how well this sequence of bits resembles a truly random sequence” (Hell et al., 2009). This section highlights some tests adopted from the statistical test suite published by the National Institute of Standards and Technology in August 2008 (Rukhin et al., 2008).

The Chi-square distribution

This is a very powerful statistical test. Its distribution can be used to compare the goodness-of-fit of the observed frequencies of events to their expected frequencies under a hypothesized distribution (Menezes et al., 1996). Figure 6.4 shows clearly that the proposed cipher passes this test. The random bits were derived from a Gaussian distribution as shown in Figure 6.5.

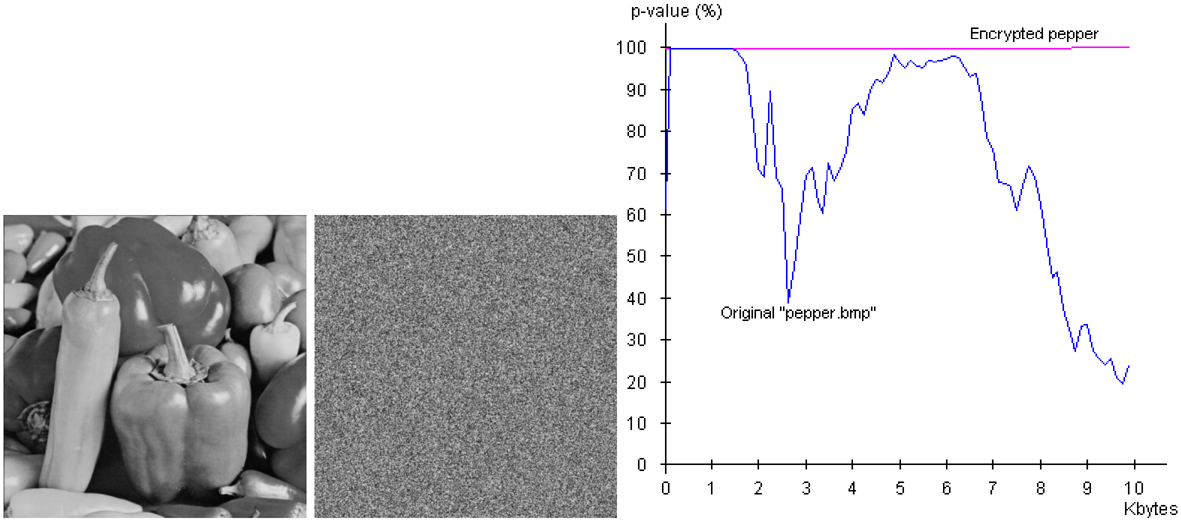


Figure 6.4: The Chi-square χ^2 distribution of the original and encrypted signals

Frequency test (monobit test)

Given a randomly generated N-bit sequence, it is expected that approximately half the bits in the sequence map to ones and approximately half of the bits map to zeros. The frequency test checks that the number of ones in the sequence is not significantly different from N/2 (Kanso & Smaoui, 2009). It is noticed that the complex imaginary part of the Fast Fourier Transform exhibits conjugate symmetry. Figure 6.5 exemplifies such a property where the magnitude of the transform is centred on the origin $\mathrm{imag}(f(u,v)) = 0$.

Shown in Figure 6.5 are: (left) the imaginary part of $f(u,v)$, in Equation (4.1) in Chapter 4, asserting that $P(\mathrm{imag}(f(u,v)) > x) = P(\mathrm{imag}(f(u,v)) < -x)$, for any *x* value (right) the corresponding binary map after applying the threshold as discussed in Chapter 4. The number of non-zero matrix elements is 32766~=N/2, where N= 256x256=65536. In other words, Equation (4.2) yields a balanced binary sequence which passes this test. This assertion holds true for any 8-bit image as well as binary images.

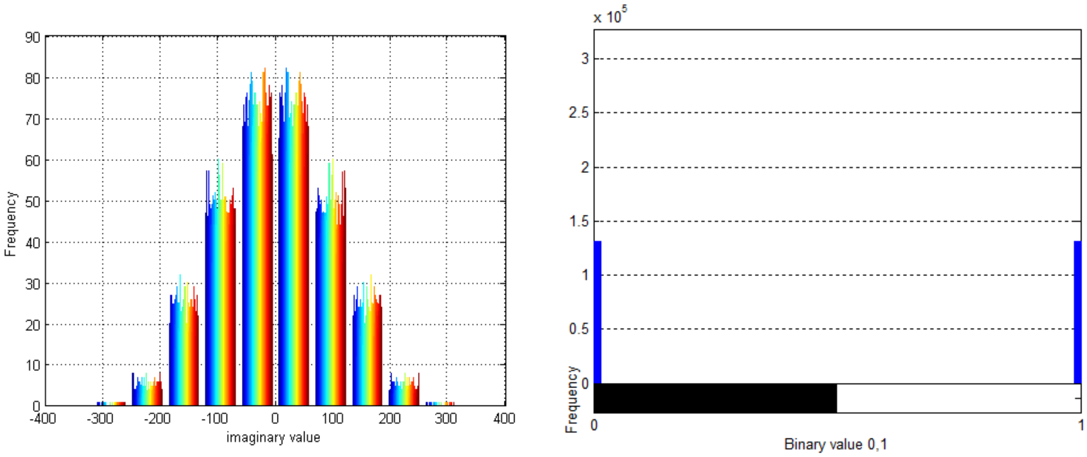


Figure 6.5: Overcoming the frequency test. (Left) imaginary part of the FFT encrypted signal and (right) the thresholded signal giving the an equal value distribution

Let the length of the encrypted bit string be *n* and let the generated bit sequence be given as $\varepsilon = \varepsilon_1, \varepsilon_2, ..., \varepsilon_n$.where : $\varepsilon_i \in \{0,1\}$. This sequence is summed up in the following manner: $S_n = X_1 + X_2 + ... + X_n$, where : $X_i = 2\varepsilon_i - 1 = \pm 1$. The P-value can be computed using the complementary error function, *erfc*, as shown in Equation (6.1).

$$P - \text{value} = \text{erfc}\left(\frac{|S_n|}{\sqrt{2n}}\right) \tag{6.1}$$

Testing this on the image “pepper_encrypted.bmp” yields $P - \text{value} = \text{erfc}(0.3950 / \sqrt{2}) = 0.6928$. Since the P-value is ≥ 0.01, decision rule at the 1% level, common values of α in cryptography are approximately 0.01 (Rukhin et al., 2008), then the bit sequence is accepted as random.

Another test is conducted on the image shown in Figure 6.6. The figure shows (left to right) the original image demonstrating different smooth blocks, the encrypted image using AES, implementation of (Buchholz, 2001), and also using the proposed method, respectively.

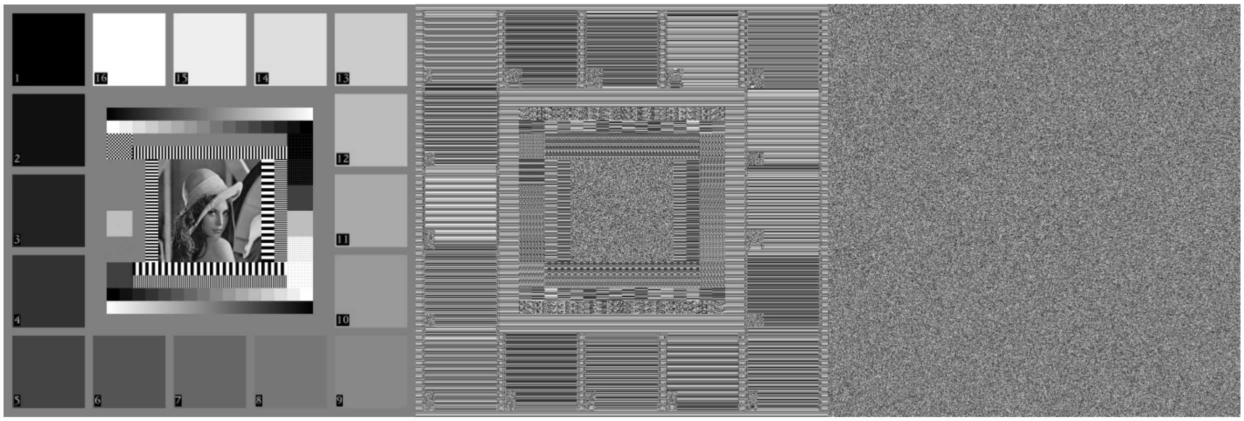


Figure 6.6: Performance of proposed method against AES in confusing image structure. Original image (left), encrypted with AES (middle) and with our proposed method (right)

As can be seen in Figure 6.6 the proposed algorithm performs better than AES in confusing the structure of the image content and also in generating the needed balanced bit stream, see,Table 6.2 and Figure 6.7. Table 6.2 shows the number of 1s indicated by PNZ_n for the proposed method and $AESNZ_n$ for the AES algorithm across all bit plans (1st-8th). The table also shows the number of zeros indicated by PZ_n for the proposed method and $AESZ_n$ for the AES algorithm across all bit plans (1st-8th).

Table 6.2: Monobit test, the proposed method against AES, used to construct Figure 6.7

Bit plan\method	*Proposed*		*AES*	
	PNZ_n (*)	*PZ_n*	*$AESNZ_n$*	*$AESZ_n$*
1st	524741	523835	519587	528989
2nd	524678	523898	516426	532150
3rd	524061	524515	523456	525120
4th	524968	523608	500456	548120
5th	523821	524755	534373	514203
6th	523118	525458	485999	562577
7th	523248	525328	497225	551351
8th	524838	523738	555971	492605

* Z_n : Number of zeros and NZ_n: Number of non-zeros, i.e., 1s

The plot in Figure 6.7 is derived from calculating the absolute value of the difference of PNZ_n and PZ_n shown inTable 6.2. This justifies the unsuitability of AES algorithm in encrypting digital images. Chaotic maps on the other hand, e.g., Logistic map, cannot guarantee the balance of each generated bit, since its variant density function is not uniform (Li et al., 2008b).

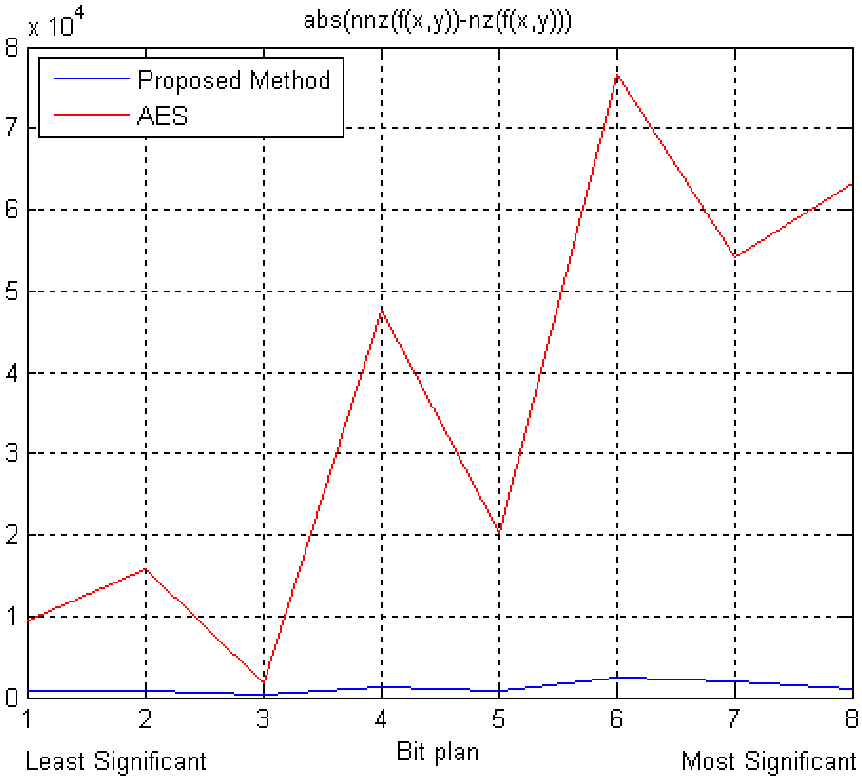


Figure 6.7: Monobit test on the encrypted images shown in Figure 6.6

Runs test

The focus of this test is on the total number of runs in the sequence, where a run is an uninterrupted sequence of identical bits. The purpose of the runs test is to determine whether the number of runs of ones and zeros of various lengths is as expected for a random sequence. In particular, this test determines whether the oscillation between such zeros and ones is too fast or too slow (Rukhin et al., 2008). Testing this on the image "pepper_encrypted.bmp" yields:

P-value = erfc((1048656-(2*2097152*0.4999*(1-0.4999)))/(2*sqrt(2*2097152)*0.4999*(1-0.4999)))

= erfc (0.0782)

= 0.9120.

The total number of runs for this example (pepper_encrypted.bmp) denoted by the value "1048656" is large enough to indicate an oscillation in the bit stream which is too fast as can be expected in a random sequence. Since the obtained P-value of 0.9120 is ≥ 0.01, the sequence is accepted as random.

Cross-covariance sequence

This test estimates the cross-covariance sequence of random processes. A natural image tends to have different randomness along its bit eight levels as shown in Figure 6.8. This figure shows (from left to right) original "pepper.bmp" 7th bit, 5th bit, 3rd bit and 2nd bit distribution, respectively. It is simply the cross-correlation of mean μ removed sequences as in Equation (6.2).

$$\phi_{\varepsilon}(\mu) = E\{\varepsilon_i - \mu\} \quad (6.2)$$

where E{.} is the expected value operator.

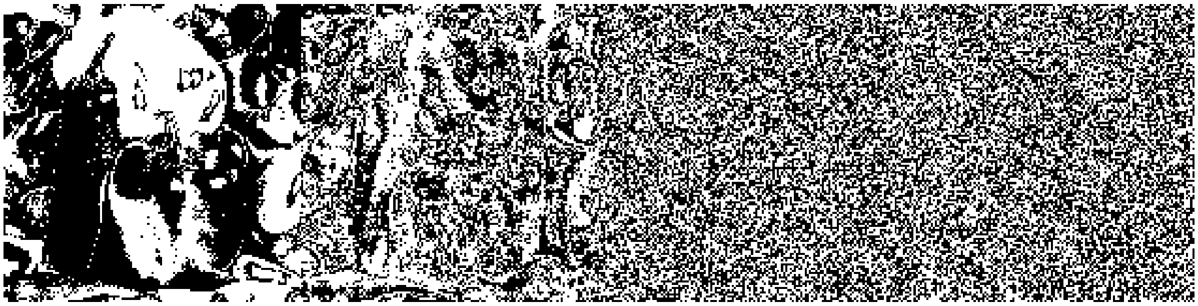
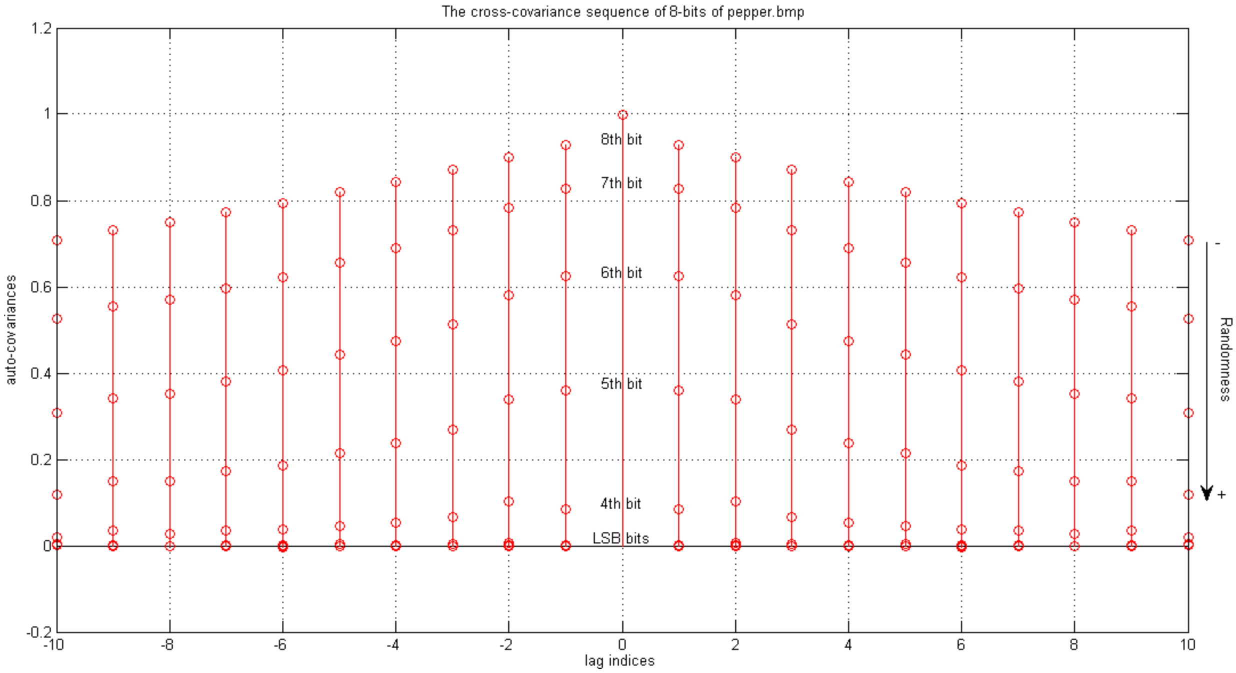


Figure 6.8: Figure showing the randomness in natural images

The sequence is further normalized so the auto-covariances at zero lag are identically 1.0, i.e., the spike appearing at zero in Figure 6.9. The figure shows: (top) projection of each bit level from the plain image "pepper.bmp" and (bottom) a great randomness shown on all bit levels of the encrypted image. This phenomenon definitely helps mimic the least significant bits when embedding the encrypted secret data.



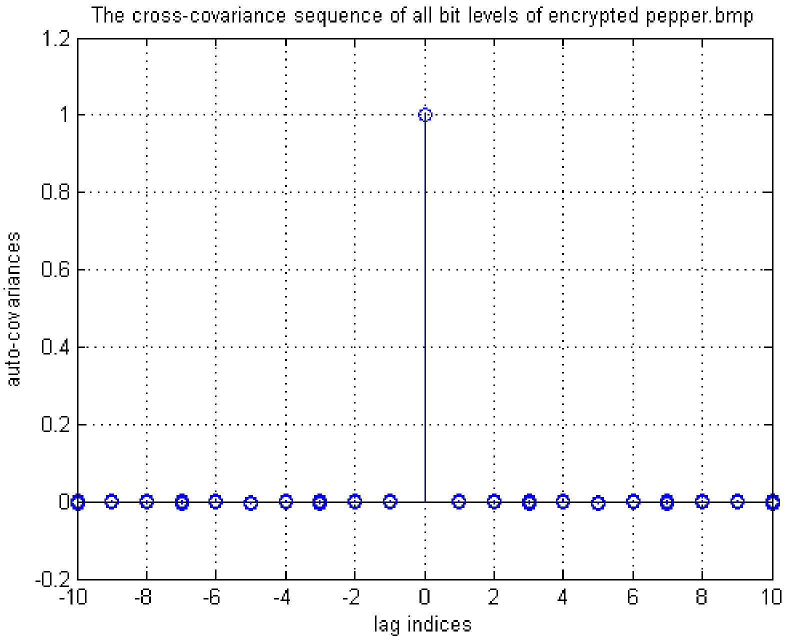


Figure 6.9: Cross-covariance test for randomness, of the original image (top) and of the encrypted version (bottom) of the eight bits. Note that in the bottom figure, data are collapsed into almost one distribution

6.1.5 Differential analysis

In order to determine the secret key, an adversary might try to establish a relationship between the plain image and its cipher version by observing the influence of a one pixel change on the overall encryption output. This kind of cryptanalysis becomes void when such a slight change results in a major transformation on the cipher. This influence is usually measured using a percentage value with the metric NPCR (Number of Pixel Change Rate). The NPCR calculates the number of pixel differences in two cipher images relating to two plain images having only one pixel difference and created using the same secret key (Wang, 2009, p.345).

Let the plain image's cipher be $\overline{A}$ and the one pixel difference generated cipher be $\overline{\overline{A}}$, then the NPCR can be obtained straightforwardly as

$$\mathrm{NPCR} = \frac{\sum_{i=1}^{H}\sum_{j=1}^{W} D_{i,j}}{W \times H} \times \ 100\% \tag{6.3}$$

where, $D_{i,j} = \begin{cases} 0 & \text{if} \quad \overline{A}_{i,j} = \overline{\overline{A}}_{i,j} \\ 1 & \text{if} \quad \overline{A}_{i,j} \neq \overline{\overline{A}}_{i,j} \end{cases}$, $1 \geq i \leq H, 1 \geq j \leq M, H$ and W denote the height and width of the image, respectively.

According to Kwok and Tang (Kwok & Tang, 2007), the expected value of NPCR of two random images is estimated by:

$$\xi(\mathrm{NPCR}) = (1 - 2^{-L}) * 100\% \tag{6.4}$$

where L corresponds to the number of bits that represent a colour component. For greyscale images L=8 bits. Hence, it is sought that $\xi(\mathrm{NPCR}) = (1 - 2^{-8}) \times 100\% = 99.60\%$. Table 6.3 contrasts the proposed method with other algorithms in terms of NPCR.

Lena and *Goldhill* images of size 512x512 were used, where the plain images used to produce $\overline{A}$ and $\overline{\overline{A}}$ have only one pixel difference as shown in Figure 6.10 and Figure 6.11.

Table 6.3: Difference between encrypted images- their plain images differ by one pixel

	NPCR (%)	
Algorithm	*Lena*	*Goldhill*
(Mitra et al., 2006)	99.6700	99.4700
(Yen & Guo, 2000)	99.4700	99.2300
(Maniccam & Bourbakis, 2004)	99.1300	99.0800
(Socek et al., 2005)	99.3300	99.2100
(Kwok & Tang, 2007)	99.6024	Not reported
(Patidar et al., 2009)	99.6094	Not reported
(Huang & Nien, 2009)	99.5200 (Average)	Not reported
(Mazloom & Eftekhari-Moghadam, 2009)	99.60937 (Average)	Not reported
(He et al., 2009)	99.6096 (Average)	Not reported
Proposed	99.6155	99.6090

Figure 6.10 shows: (top-left to right) the original image Goldhill, its encrypted version, $\overline{A}$, Goldhill with one pixel difference and its encrypted version, $\overline{\overline{A}}$, respectively. The similarities between the two encrypted images is shown as a binary image in the bottom representing $D_{i,j}$, black pixels.

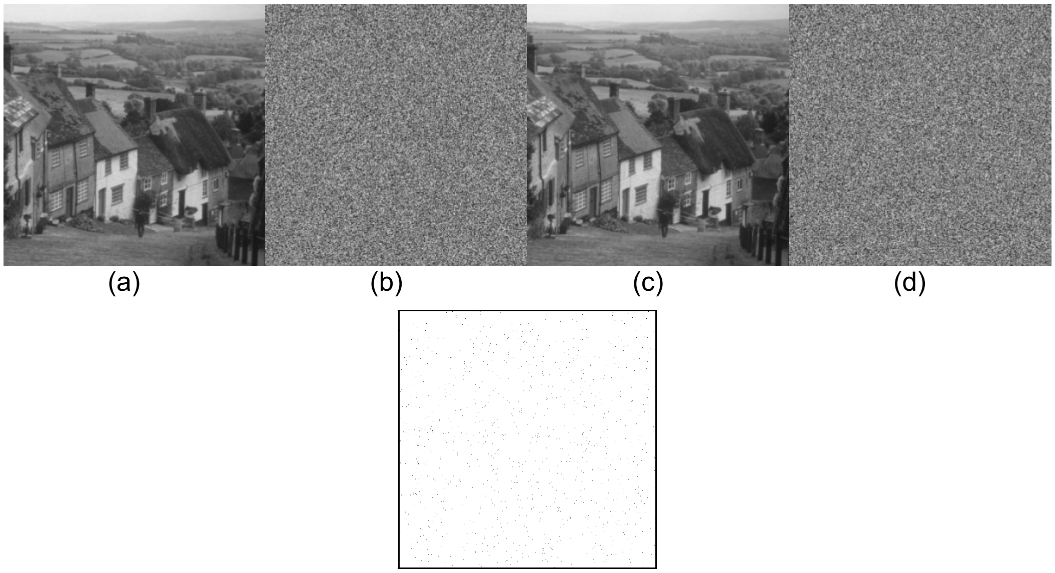
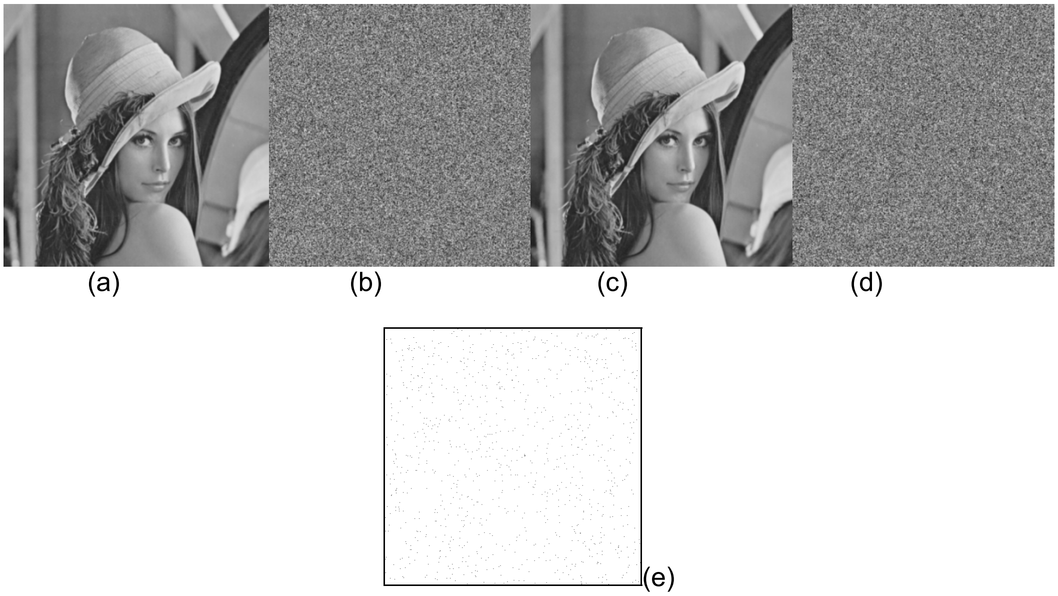

(a) (b) (c) (d)

Figure 6.10: Goldhill- differential analysis. (a-d) original image, encrypted version, same image as in *(a)* with one pixel incremented by one, its encrypted version, respectively and (bottom) the difference between (b) and (d)

Figure 6.11 shows: (top-left to right) the original image Lena, its encrypted version, Lena with one pixel difference and its encrypted version, respectively. The similarities between the two encrypted images is shown as a binary image in the bottom representing $D_{i,j}$, black pixels.



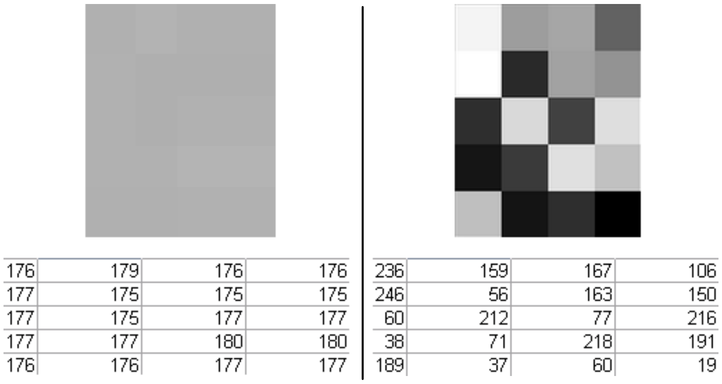
(a) (b) (c) (d)

(e)

Figure 6.11: Lena- differential analysis. (a-d) original Lena, encrypted version, same image Lena with one pixel incremented by one, its encrypted version, respectively and (bottom) the difference between (b) and (d)

6.1.6 Other security issues

Two additional aspects of the proposed method are highlighted here. The first feature is that the proposed scheme is capable of not just scrambling data but it also changes the intensity of the pixels which contributes to the safety of the encryption. For convenience, Figure 6.12 illustrates a cropped greyscale matrix of size 4x5 from a natural image along with its encrypted version. As can be appreciated from the figure, the algorithm combines confusion and diffusion. Notice how same gray values are encrypted differently. This irregularity is very important as it hampers any attempt to reverse attack the algorithm.



176	179	176	176
177	175	175	175
177	175	177	177
177	177	180	180
176	176	177	177

236	159	167	106
246	56	163	150
60	212	77	216
38	71	218	191
189	37	60	19

Figure 6.12: A 4x5 cropped plain patch from a natural image. (Left) original homogenous area with its grayscale values and (right) the encrypted version with its grayscale values

The second feature of the proposed algorithm is the unbiased handling of both gray scale and binary images. Methods involving chaos are special cases where they can be considered analogous to encryption when dealing with binary plain images.

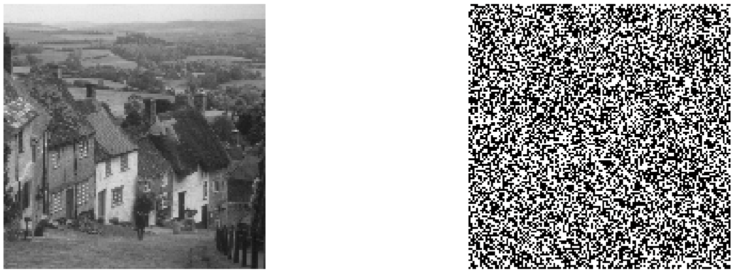
If an image contains homogenous areas large redundant data will surf and thwart the efficiency of encryption algorithms laying ground for a codebook attack. This is due to consecutive identical pixels which lead to the same repeated patterns when a block cipher is used in the Electronic Code Book, ECB, mode (Shujun et al., 2004). Since the proposed algorithm is not block based, therefore, this kind of phenomenon does not occur.

An attacker cannot work backwards to deduce previous random values by observing the internal state of the algorithm. Attackers can also use computer clusters to break encrypted strings by predicting the output until enough entropy is obtained. This might work on text encryption using a dictionary attack, but as far as digital imaging is concerned, the computational prediction of such entropy that mimics the human visual system, HVS, is complex and vague, therefore its feasibility is questionable.

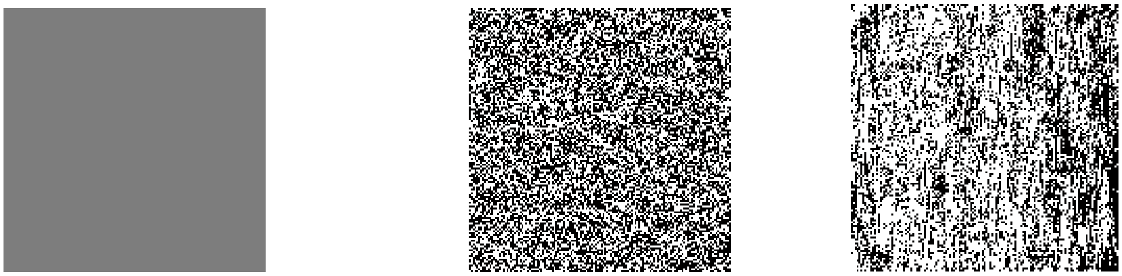
The Chosen Plaintext Attack, CPA, is an attack model in which an attacker is presumed to have the ability to encrypt a plain image to obtain its corresponding cipher. The purpose of this attack is to exploit weaknesses in the encryption algorithm in the hope of revealing the scheme's secret key as shown in Equation (6.5).

$$A = A' \otimes (B' \otimes \mathrm{Map}) \tag{6.5}$$

where A is the decrypted image, A' its cipher, B' is the attacker's encrypted neutral image, Figure 6.13(c), $\otimes$ is the XOR operation and Map is the key, $B' \otimes \mathrm{Map}$ is shown in Figure 6.13 (d). Note that Figure 6.13 (a) and Figure 6.13 (c) were encrypted with the same encryption key, to simulate the worst case of attack, and then Equation (6.5) is applied to yield Figure 6.13 (e)Figure 6.13. This scenario presumes that the attacker knows *(c)* and the encrypted version of *(c)* using the same secret key that encrypted *(a)*. The attacker's task here is to derive a key map with which he can decrypt *(b)* to yield *(a)*. The proposed algorithm bypasses this test as shown in (e) which is $b \otimes d(\mathrm{Map})$.



(a) Original image (b) 7th bit plan of the encrypted image of (a)



(c) Neutral uniform image used for attack (d) Derived key map (e) Xor(d,b)

Figure 6.13: CPA cryptanalysis attack

Random Control Encryption Subsystem (RCES) algorithm, which is a chaos-based encryption scheme, is proven weak against this attack as confirmed in (Li et al., 2008b).

The results so far demonstrate that the proposed encryption algorithm is superior to the work of Pisarchik (Pisarchik et al., 2006) in terms of algorithm complexity and parameter requirements. Moreover, the algorithm is securely backed up by a strong 1D hash function.

In (Pisarchik et al., 2006) the desired outcome converges after some iterations which needs to be visually controlled to flag the termination of the program. However, in this work the algorithm is run only once for each colour component, *R*, *G* and *B*. The proposed method needs only one input from the user, which is the password, and it will handle the rest of the process, while in (Pisarchik et al., 2006) three parameters are required. The proposed method obviously can be applied to gray scale images as well as binary images. These extensions are not feasible in (Pisarchik et al., 2006) as they incorporate into their process relationships between the three primary colours, *R*, *G* and *B*. Finally, time complexity which is a problem admittedly stated in (Pisarchik et al., 2006) would be reduced greatly by adopting this work's method. The algorithm is coded using MATLAB, which is an interpreted language, and Pisarchik (Pisarchik et al., 2006) used C#.

Steganoflage is tested on the same test image as in (Pisarchik et al., 2006) to establish a fair judgement. To demonstrate visually the confusion requirement being met, Figure 6.14 illustrates this test. Even though only a small change has occurred, the final two ciphers differ dramatically as can be seen from Figure 6.14(d). The proposed algorithm shows better performance compared to other recent methods such as the works in (Zeghid et al., 2006), (Zou et al., 2005), (Wong et al., 2008b), (Wong et al., 2008a), (Lian et al., 2005) and (Wang & Zhang, 2008), in addition to the conventional PRNG, see Table 6.1.

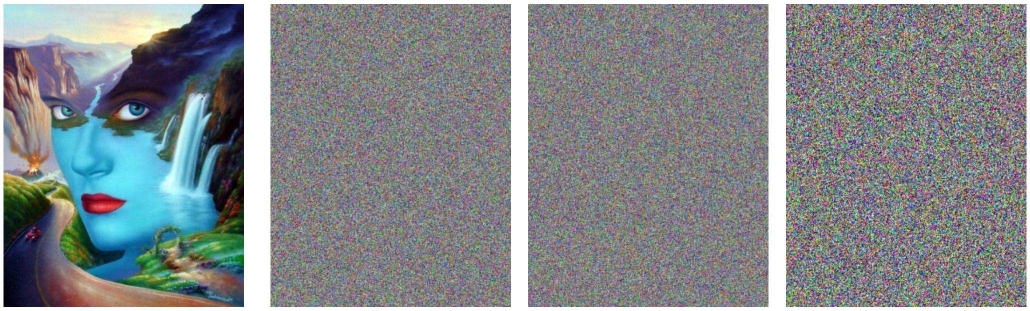

(a) (b) (c) (d)

Figure 6.14: key sensitivity test for colour images. (a) test image, Mother of the Nature, (b) cipher using 'Steganography' as a password, (c) cipher using 'Steganographie' as a password and (d) the difference between *(b)* and *(c)*.

Pisarchik (Pisarchik et al., 2006) measured the contrast between two given images using an image histogram. Even though an image histogram is a useful tool, unfortunately, it does not tell us much about the structure of the image or about the displacement of colour values. Histograms accumulate similar colours in distinguished bins regardless of their spatial arrangement. A better alternative would be to use similarity measurement metrics such as the popular PSNR, which is classified under the difference distortion metrics.

Table 6.4 compares the PSNR values showing further information on the diffusion/confusion aspects. It is mentioned in sub-section 6.1.6 that Pisarchik's algorithm (Pisarchik et al., 2006) involves a rounding operator applied each time the program is invoked by the different iterations. This feature is not adopted as there will be a loss of information when the embedded data is reconstructed.

Table 6.4: PSNR values of the different generated ciphers

Chaos	Figure 6.14 (a)	Figure 6.14(b)	Figure 6.14(c)
Figure 6.14 (a)	-	7.8009 dB	7.8010 dB
Figure 6.14(b)	-	-	7.7765 dB

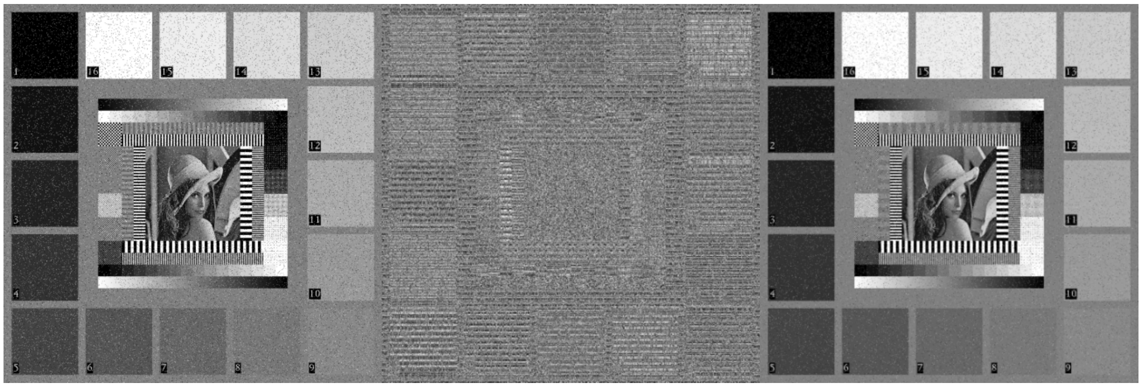
Table 6.5 shows the advantages of using the proposed encryption method in comparison with other methods particularly in steganography applications.

Table 6.5: Comparison with different image encryption methods

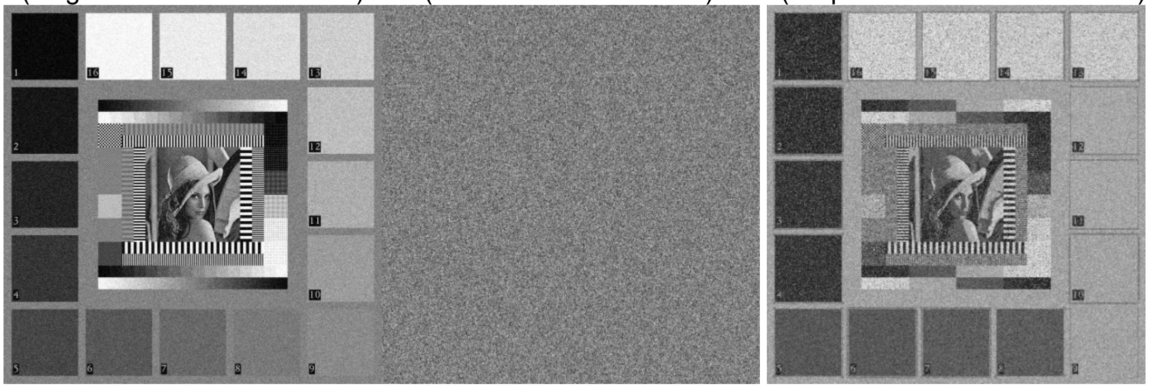
Method	*Encryption issue*	*Steganography issues*		
	Security	*Balanced bit distribution*	*Tolerance to transmission faults*	*Suitability for image coding*
AES/IDEA	excellent	weak,Figure 6.7	weak	average(*)
Chaos	average, see (Cokal & Solak, 2009)	weak	average	very good
Bit stream ciphers	weak	very good	very good	very good
Proposed	good	very good	very good	very good

(*) If the message is encrypted with a block cipher and a given block size, the lengths of embedded data varies in multiples of this unit (Westfeld, the CRYSTAL project). See also, (Patidar et al., 2009), (Chen & Zheng, 2005), (Shujun et al., 2004) and (Hu & Han, 2009) and Figure 6.15.

The inherent properties of the proposed encryption method are helpful in obtaining a better trade-off between robustness and security. When encrypted data is transmitted through a network, errors might occur. Consequently, a tolerance to transmission faults is desirable. An obvious simulation of such faults can be achieved using added noise. Subsequently, two types of noise are examined on the decryption performance. Figure 6.15 shows deciphered images of Figure 6.6 after adding “Salt & Pepper” uniform noise with 0.05 density (top) and Gaussian white noise of zero mean with 0.01 variance. It is very obvious that AES has a stronger avalanche property, which produces poor decryption if some bits are flipped during the transfer (Socek et al., 2007). Figure 6.15 shows: (top) decrypted images after “Salt & Pepper” noise is added to the encrypted images of AES and the proposed method and (bottom) decrypted images after Gaussian white noise is added to the encrypted images of AES and the proposed method. Also shown is the original image after applying the two types of noise for mere comparison. In the course of analyzing the behaviour of the proposed encryption scheme when noise is added, some will argue that the outcome would be the same as if the noise had been added to the original image. This is not true since the algorithm is not only a result of a simple XOR operation.



(Original: PSNR=17.7548dB) (AES: PSNR=7.5814dB) (Proposed: PSNR=20.6725dB)



(Original: PSNR=20.6185dB) (AES: PSNR=7.6415dB) (Proposed: PSNR=12.8455dB)

Figure 6.15: Noise attack. (Left column) noise applied directly to original image, top: “Salt & Pepper” and bottom: Gaussian noise, (middle column) decrypted image after applying noise to the encrypted AES image, top: “Salt & Pepper” and bottom: Gaussian noise and (right column) decrypted image after applying noise to the proposed encrypted image, top: “Salt & Pepper” and bottom: Gaussian noise

6.2 Evaluation of Skin Tone Detection Algorithm

Since in steganography the choice of cover images is not rigid, a decision is made to target images with human presence for the following reasons:

- identifying human-skin regions is fast and invariant to translation, rotation and scaling
- human-skin areas in digital images exhibiting moderate texture provides a fast automatic mechanism to avoid smooth areas for embedding
- skin-tone areas can be altered without significantly impairing the quality of image perception as such areas provide information that are psycho- visually redundant (Gonzalez & Woods, 2002, p.417)

- by recognizing skin-tone areas an image can be divided into two main clusters, one could be used for embedding, skin-tone areas, and the other used to correct any statistical distortions
- the chrominance, C_r and C_b in the YC_bC_r colour space, of human skin-tone is always in the middle range which provides immunity to the overflow and underflow problem in wavelets reconstruction

For unconvincing reasons, illumination was abandoned by researchers who instead tackled the problem of skin colour detection thinking such a channel had no relevant information for extracting and classifying skin colour pixels. It will be shown that illumination involvement can significantly increase the robustness of the detector. However, like all existing algorithms, it is not yet intelligent enough to discriminate whether a scene possesses a skin colour or something that looks similar. The proposed colour model and the classifier can cope with difficult cases encapsulating bad and uneven lighting distribution and shadow interferences. Consequently, these results respond evidently to those authors who arguably questioned the effectiveness of the use of illumination based on its inherent properties. The proposed algorithm outperforms both YC_bC_r and NRGB which have attracted many researchers to date.

Figure 6.16 exemplifies how inherent properties of luminance can aid performance if handled intelligently. Shown in the figure are: (left to right) original input image, image 8 in Table 6.6, skin tone detected by (Hsu et al., 2002), by (Berens & Finlayson, 2000) and by the proposed method in this work respectively. Notice how the proposed colour space is not affected by the colour distribution which enabled *Steganoflage* to detect skin tone with better efficiency.

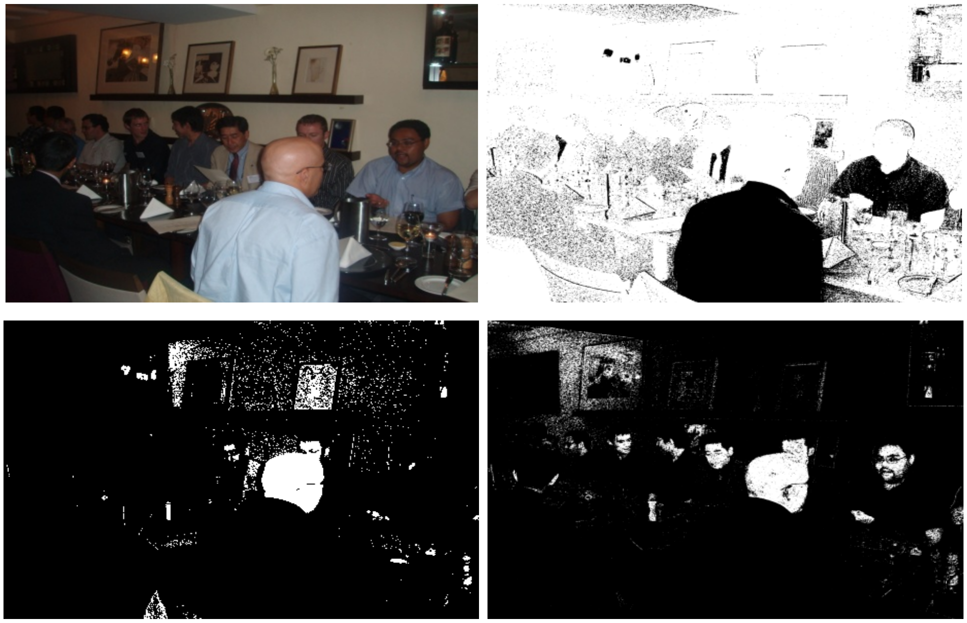


Figure 6.16: Skin detection in an arbitrary image, the proposed method shown on the bottom right

Figure 6.17 shows the test images from a collection of images downloaded from Internet and the corresponding detected skin regions of each algorithm. In the figure are shown: (left column to right) original images with outputs of (Hsu et al., 2002), (Berens & Finlayson, 2000) and (Kovač et al., 2003) and the proposed method, respectively.

The images in Figure 6.17 are samples taken from the Internet database that appears in Table 6.6, where the top corresponds to image 1 and the bottom to image 2 in the table. Other samples exhibiting dark skin colour are shown in Figure E.1 (Appendix). As shown, the proposed algorithm is insensitive to false alarms. Therefore, it has the least false negative pixels compared to the other three methods, which renders the output cleaner in terms of noise interference.

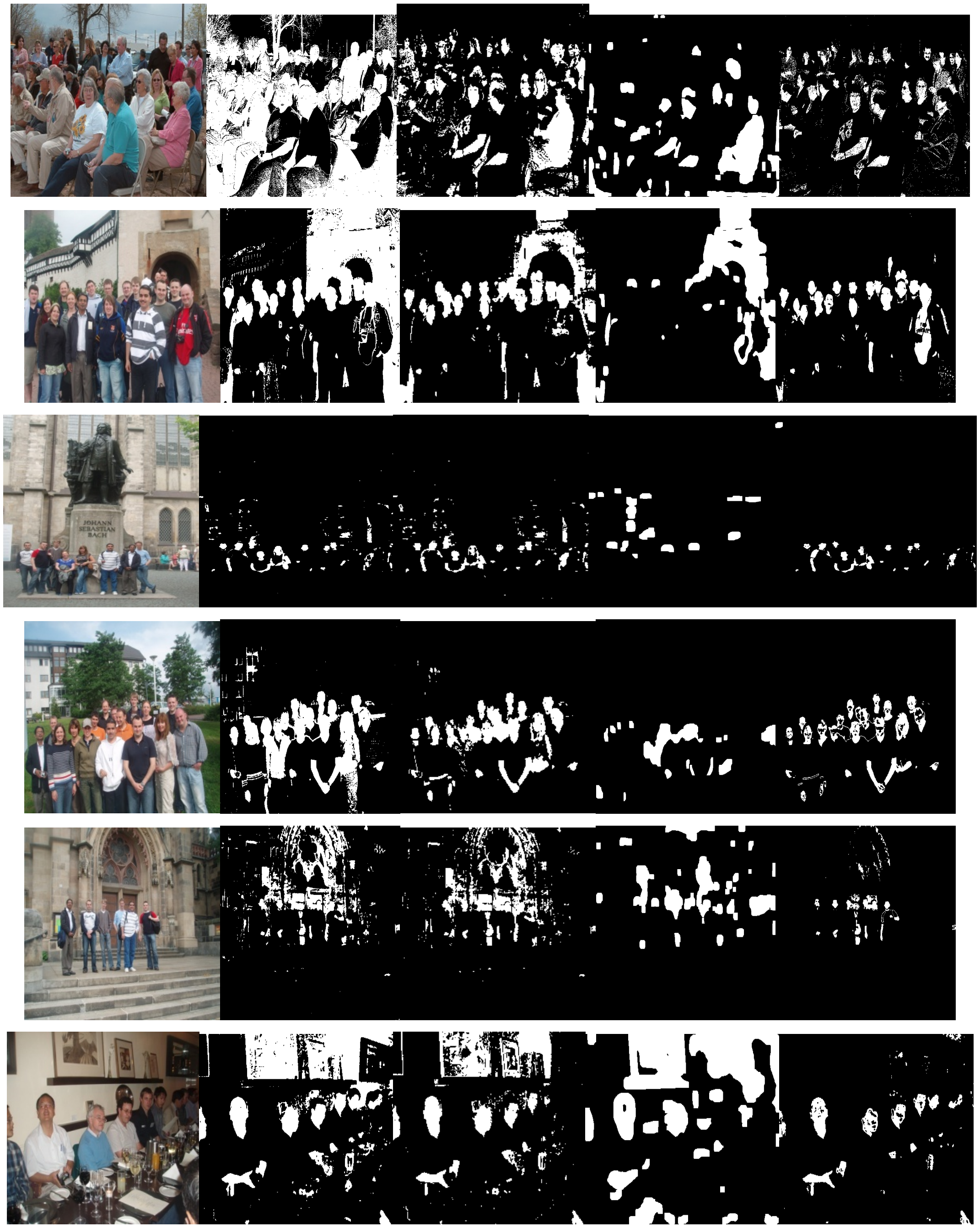


Figure 6.17: Performance analysis of skin tone detection on arbitrary images. (Left column to right) Original image, (Hsu et al., 2002) method, (Berens & Finlayson, 2000) method, (Kovač et al., 2003) method and the proposed method, respectively

The ultimate advantage that the proposed method offers is the reduction of dimensionality from 3D to 1D, which contributed enormously to the algorithm's speed as can be seen in Table 6.7 where the proposed method is compared against other methods (Hsu et al., 2002), (Berens & Finlayson, 2000) and (Kovač et al., 2003) on 12 images obtained from the Internet database of which samples are shown in Figure 6.17.

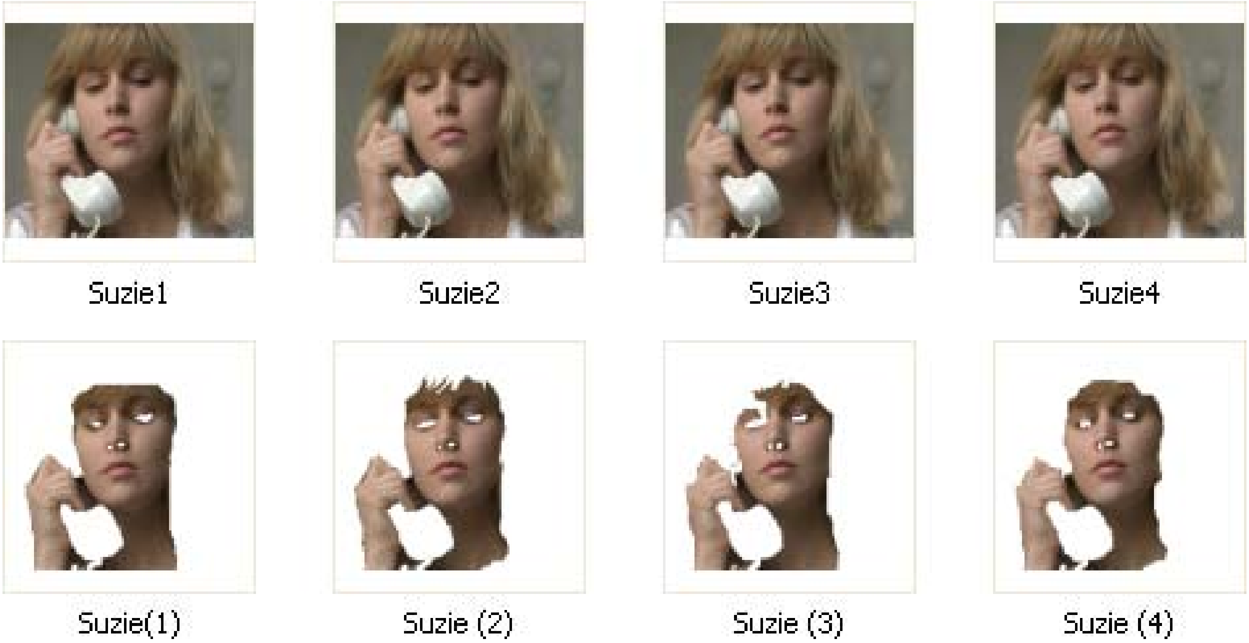
Table 6.6: Comparison of computational complexity

Image #	*Number of Pixels*	*Time elapsed in seconds*			
		(Hsu et al., 2002)	(Berens & Finlayson, 2000)	(Kovač et al., 2003)	Proposed
1	840450	0.5160	33.515	7.796	0.125
2	478518	0.4060	22.094	4.156	0.047
3	196608	0.2970	4.547	2.188	0.062
4	196608	0.3280	3.563	1.906	0.062
5	849162	0.5160	33.062	7.531	0.078
6	850545	0.6090	39	8.343	0.062
7	849162	0.6090	39.219	6.641	0.078
8	849162	0.5160	39.172	8.484	0.078
9	849162	0.6100	38.203	6	0.078
10	7750656	3.1720	> 600 *	54.86	0.562
11	982101	0.6410	79.469	7.297	0.078
12	21233664	9.3910	> 600 *	144	1.531

(*) the Log algorithm [25] did not converge after 10 min which forced us to halt its process.

These results were obtained using an Intel Pentium Dual Core Processor CPU with Memory Dual-Channel 1024 MB (2x512) 533 MHz DDR2 SDRAM and 1.6 GHz and by using MATLAB Ver. 7.0.1.24704 with IP Toolbox Ver. 5.0.1. It can be seen in Table 6.6 that the computational time required by some other methods depends on the processed image's content as the processing time is different for images even though they have the same dimensions.

In addition to the arbitrary still images from the Internet, the algorithm is tested on a larger benchmark, i.e., 150 image frames from the popular video "Suzie.avi". This movie sequence is chosen to test for the confusion that hair may cause. Depicted in Figure 6.18 are some frame samples and the hand labelled ground truth models, (top) shows original extracted frames, (bottom) the corresponding Ground Truth from the 150 manually cropped frames.

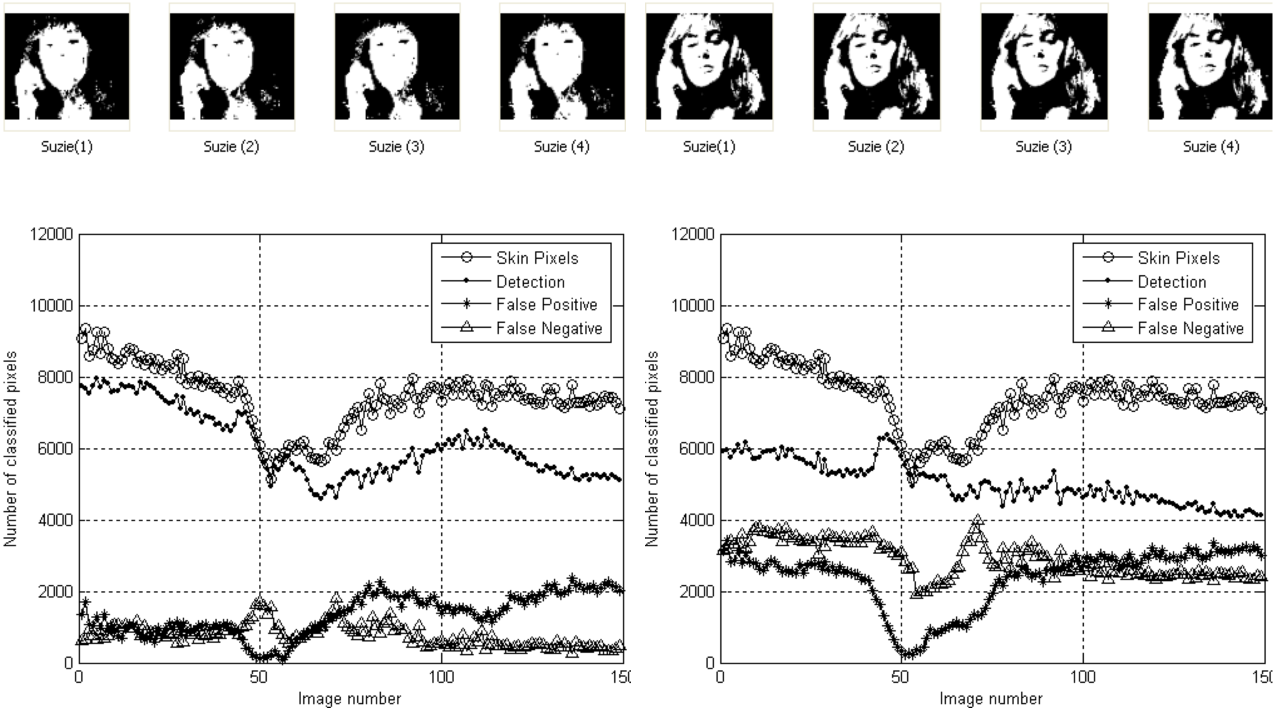


Suzie1 Suzie2 Suzie3 Suzie4

Suzie(1) Suzie (2) Suzie (3) Suzie (4)

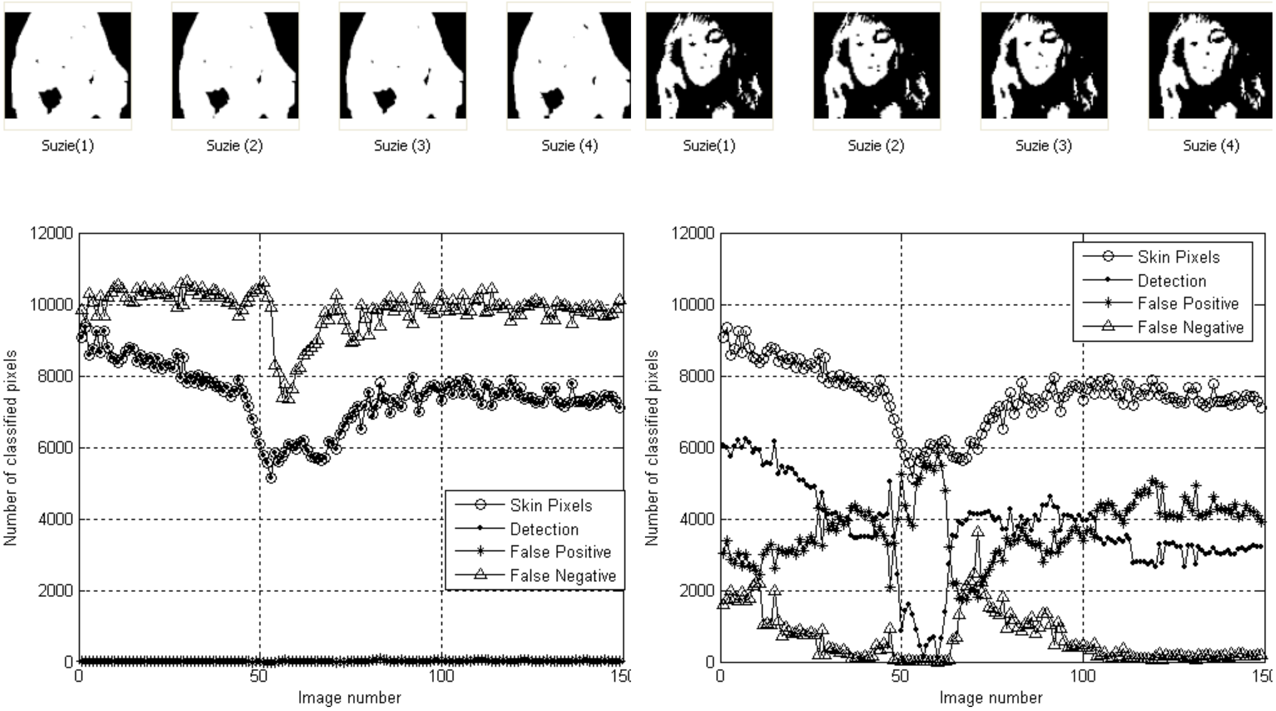
Figure 6.18: The first four frames from a standard testing video sequence. (Top) original image frames and (bottom) hand-labelled skin area

Figure 6.19 shows the first four frames and the graphical performance analysis of the proposed method against those reported methods on the entire 150 frames. As can be seen, the proposed method is by far the most efficient in that it preserves lower rates for the dual false ratios while securing a high detection rate among all methods, see, Figure 6.19.



(Proposed)

(Kovač et al., 2003)

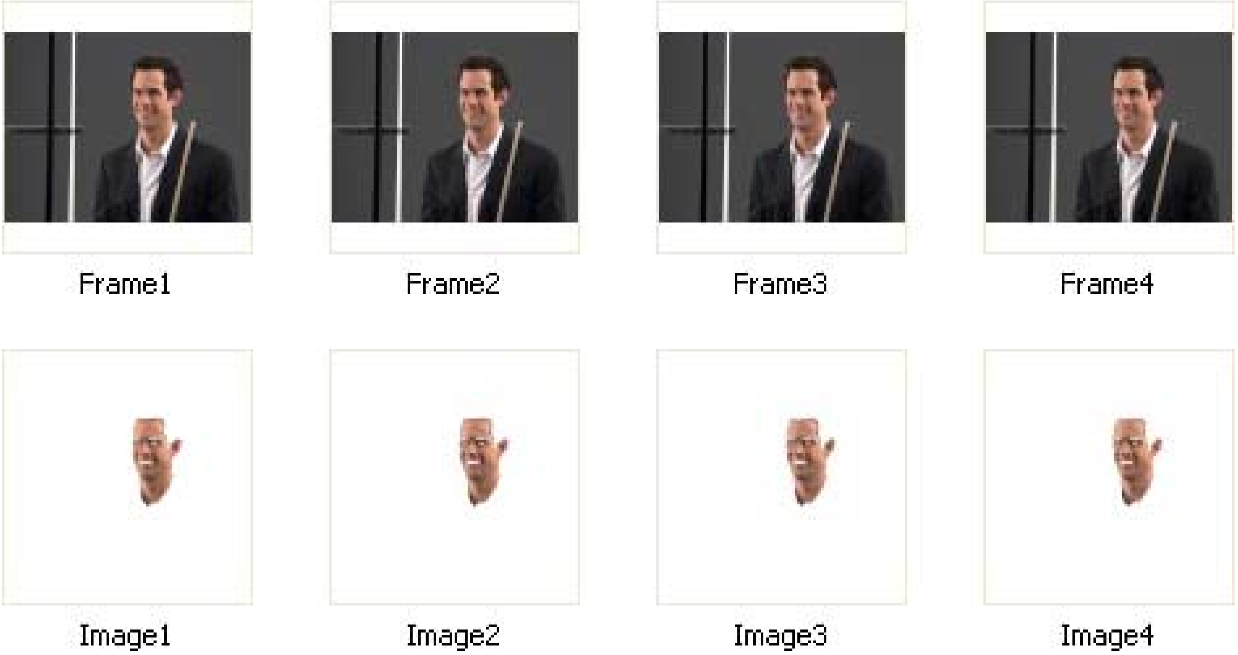


(Berens & Finlayson, 2000)

(Hsu et al., 2002)

Figure 6.19: Performance comparison of different methods “Suzie.avi”

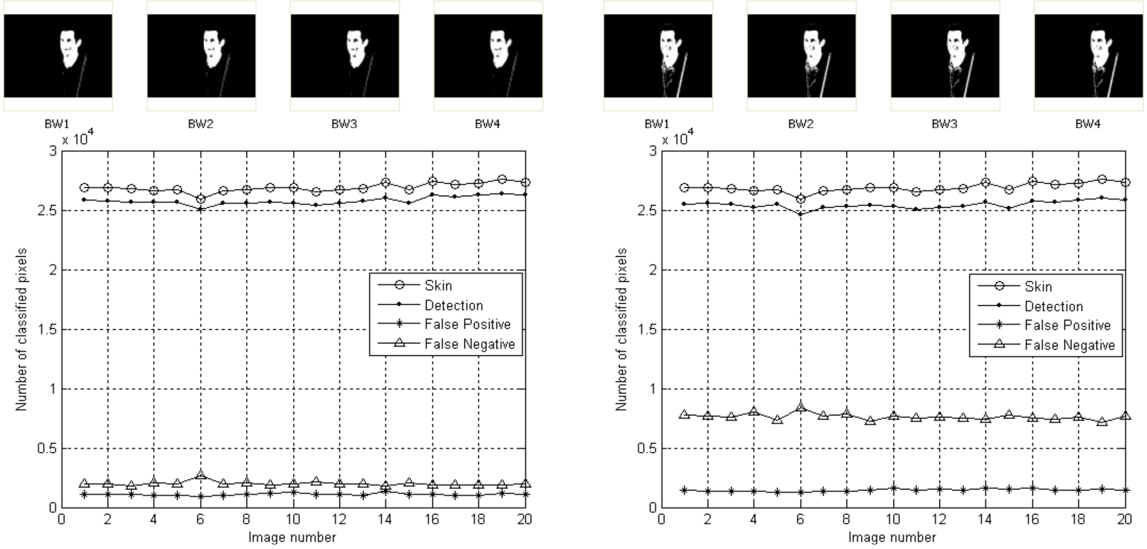
Figure 6.20 shows the first four, hand labelled, frames from “Sharpness.wmv”: (top) original extracted frames, (bottom) the corresponding Ground Truth and the overall performance is demonstrated in Figure 6.21. The video file is used by Windows Media Centre to calibrate the computer monitor by modifying frame sharpness which is suitable for testing the consistency in the performance of the algorithm.



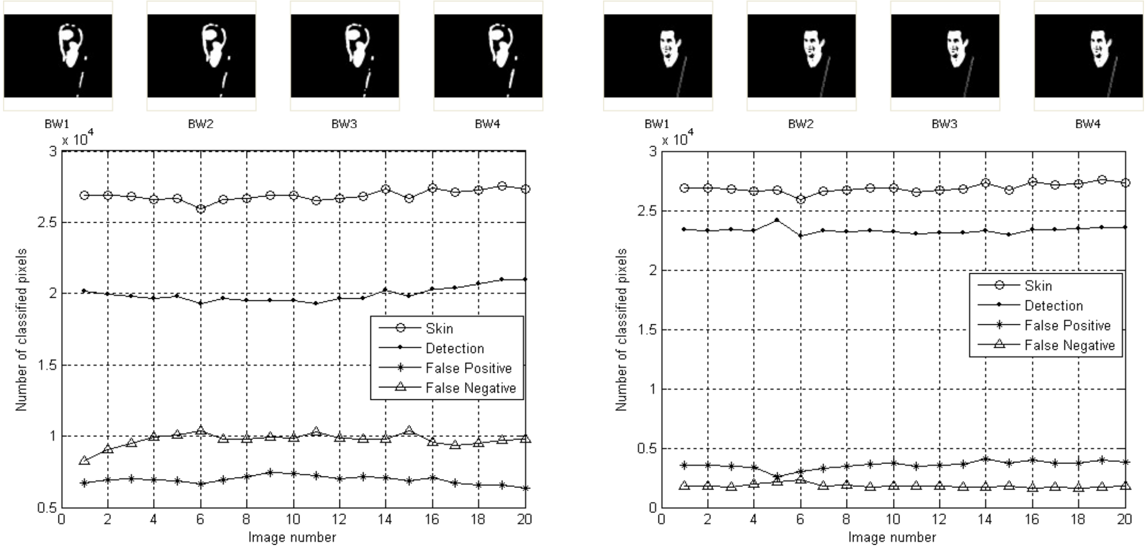
Frame1 Frame2 Frame3 Frame4

Image1 Image2 Image3 Image4

Figure 6.20: The first four frames from a Dell™ video sequence, (top) original image frames and (bottom) hand-labelled skin area



(Proposed) (Kovač et al., 2003)



(Berens & Finlayson, 2000) (Hsu et al., 2002)

Figure 6.21: The first four frames and performance analysis on “Sharpness.wmv”

6.3 Overall Robustness of *Steganoflage*

This section discusses the robustness of *Steganoflage* to statistical and visual attacks and compares its performance with other related algorithms. It also discusses some limitations of the proposed method.

6.3.1 Robustness against intentional and passive attacks

No prior work has discussed the application of skin tone detection in conjunction with adaptive steganography. All of the prior steganographic methods suffer from intolerance to any kind of image manipulation applied to the stego-image such as a Warden passive attack scenario (Siwei, 2005, p.67). Scholars differ about the importance of robustness in steganography system design. In (Cox, 2009), Cox regards steganography as a process that should not consider robustness as it is then difficult to differentiate from watermarking. Katzenbeisser, on the other hand, dedicated a sub-section to robust steganography. He mentioned that robustness is a practical requirement for a steganography system. “Many steganography systems are designed to be robust against a specific class of mapping.” (Katzenbeisser, 2000). It is also rational to create an undetectable steganography algorithm that is capable of resisting common image processing manipulations that might occur by accident and not necessarily via an attack. JPEG would be a good example for such an attack, see results in Figure 6.22.

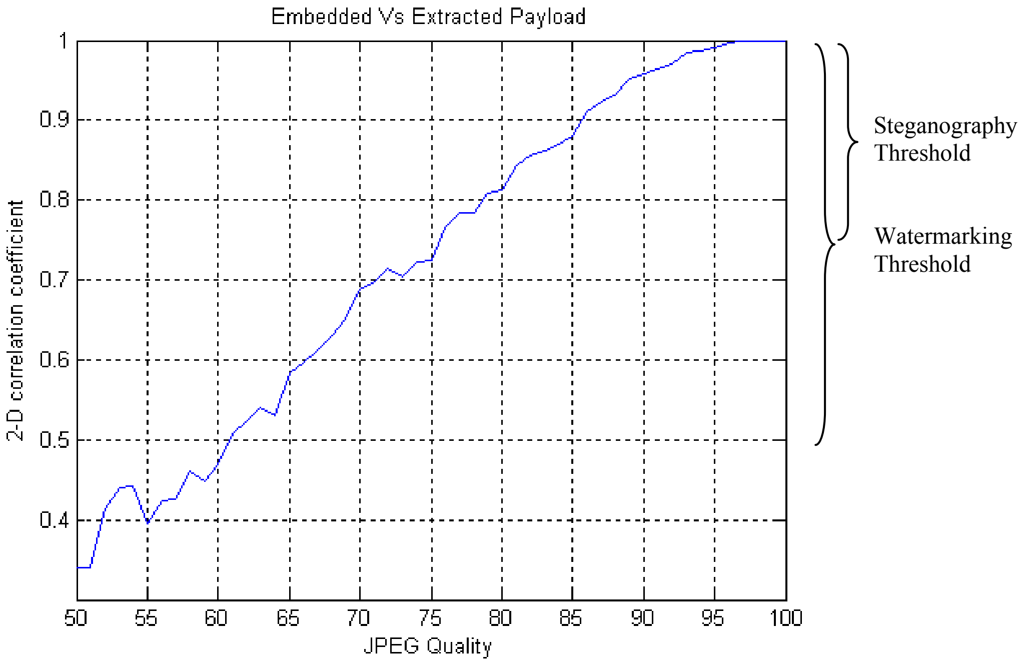
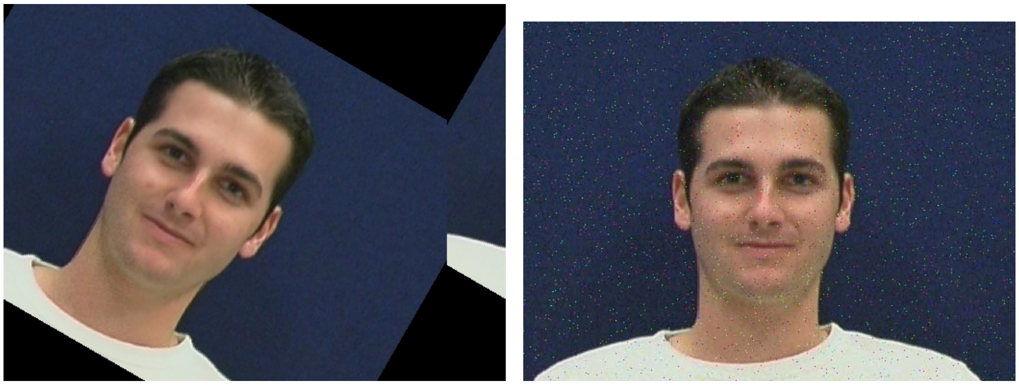


Figure 6.22: JPEG compression attack on the stego-image. The figure shows the quality of the extracted payload after applying different levels of JPEG compression. Watermarking can accept a threshold of 50% by using cross-correlation method, while steganography accepts the payload if it is visually perceptible, e.g., 75%

Three types of attacks are carried out, namely noise impulses, rotation and cropping attacks. Shown in Figure 6.23 are: (left) attacked stego-image with joint attacks of cropping, JPEG compression, and translation with an offset=60 and rotation of -30 degrees and the extracted secret data, the little error in the extracted signal is due to interpolation operation (right) attacked with salt and pepper noise and the extracted secret data, (bottom, left to right) successful extraction of embedded data after JPEG compression with quality factors Q=100, Q=80 and Q=75, respectively.

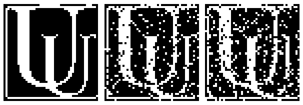


(a) stego-image attacked with rotation and translation

(b) stego-image attacked with noise

(c) extracted secret data from (a)

(d) extracted secret data from (b)



(e) extracted secret data after applying JPEG compression with Q=100%, 80% and 75% respectively

Figure 6.23: Resistance to natural image processing attacks

Figure 6.24 (top left) shows the original cover image - *ID01_035.bmp*- obtained from the GTAV Face Database along with the image annotation to embed, (top right) is the attacked stego-image and the extracted annotation, (bottom left) attacked stego-image with half transparent frame and the extracted annotation and finally (bottom right) shows an attack on stego-image with translation to the left with an offset=200 pixel and the extracted annotation which is identical to the embedded one.

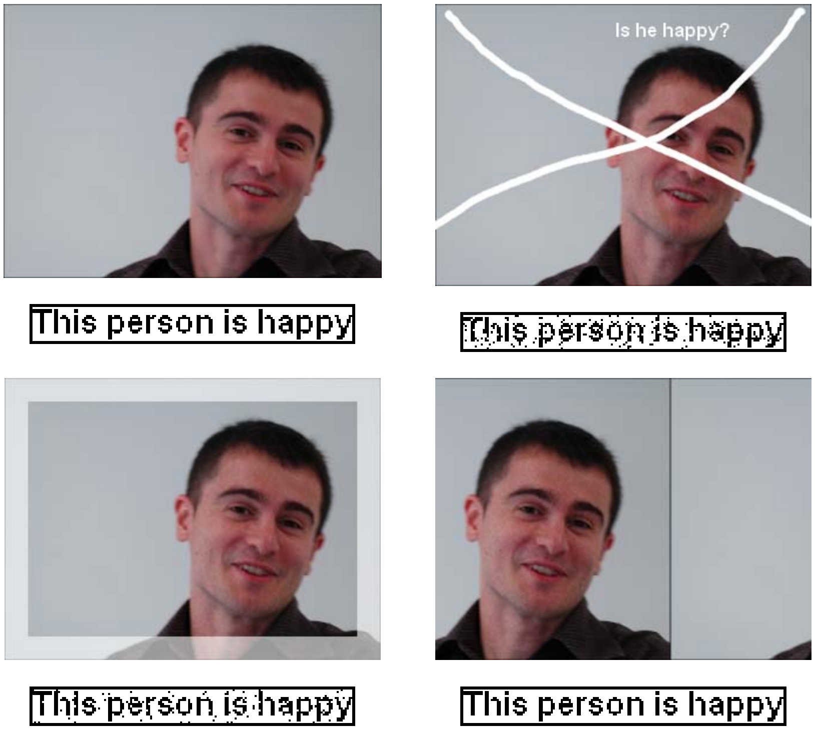


Figure 6.24: Resistance to other deliberate image processing attacks

The algorithm is capable of surviving JPEG compression attacks up to 75%, below which the hidden data will be totally destroyed. Surmounting JPEG compression is believed to be enhanced by the encryption of the payload since encryption often significantly changes the statistical characteristics of the original multimedia source, resulting in much reduced compressibility (Mao & Wu, 2006). This resilience to attacks is deemed to be essential in image steganography or watermarking. In this case, the algorithm performs better than various algorithms such as Peng's algorithm (Peng & Liu, 2008).

6.3.2 Steganalysis and visual perceptibility

In the frequency domain, Pevny and Fridrich (Pevny & Fridrich, 2007) developed a multi-class JPEG steganalysis system that comprises DCT features and calibrated Markov features, which were then merged to produce a 274-dimensional feature vector. This vector is fed into a Support Vector Machine, SVM, multi-classifier capable of detecting the presence of Model-Based steganography, F5, OutGuess, Steghide and JPHide&Seek. Initially, it was proposed a reduction of the complexity of the 274-D vector should be carried out by retaining only the most contributing features using Principal Component Analysis, PCA. A decision was made to not proceed in that direction as some authors comment against such procedures (Kodovsky & Fridrich, 2008b). Features were created derived from 200 images demonstrating different structural complexities obtained using various digital camera models in addition to images downloaded from the Internet. Another 200 stego-images were generated using the same set and also their features were similarly obtained. Then a feed-forward back-propagation network was created instead of the SVM to act as a classifier feeding into it the 400 feature vectors. An independent testing set comprising 80 images was used to simulate the network. The result confirms that the proposed scheme can overcome detection using this attack. A surprising observation was that the detection rate was slightly better when the payload was small unlike when the full skin area was used. The reported detection probability is still within a random guessing range:

sim (net, Set_Small) => 36.8421%, sim (net, Set_Full)=> 31.5789%

where *sim* denotes simulation of the neural network, *net* denotes the trained neural network, Set_Small is the set of stego-images where the skin region was not fully used and Set_Full is the case where skin-tone areas were fully used. Figure 6.25 shows the stego-image along with its PSNR value , (top-left) original image and (top-right) stego-image (PSNR=42.1293 dB) where all skin area was used, (bottom-left) original and (bottom-right) stego-image, PSNR 52.738 dB, payload=2256 bits. The second attack, namely ±1steganalysis, cannot be accomplished since the embedding changes do not produce this effect, see Figure 6.26 (left) S-Tools' ±1embedding fingerprint (right) the proposed method, which contrasts our algorithm to S-Tools showing that our algorithm is not prone to this kind of attack.

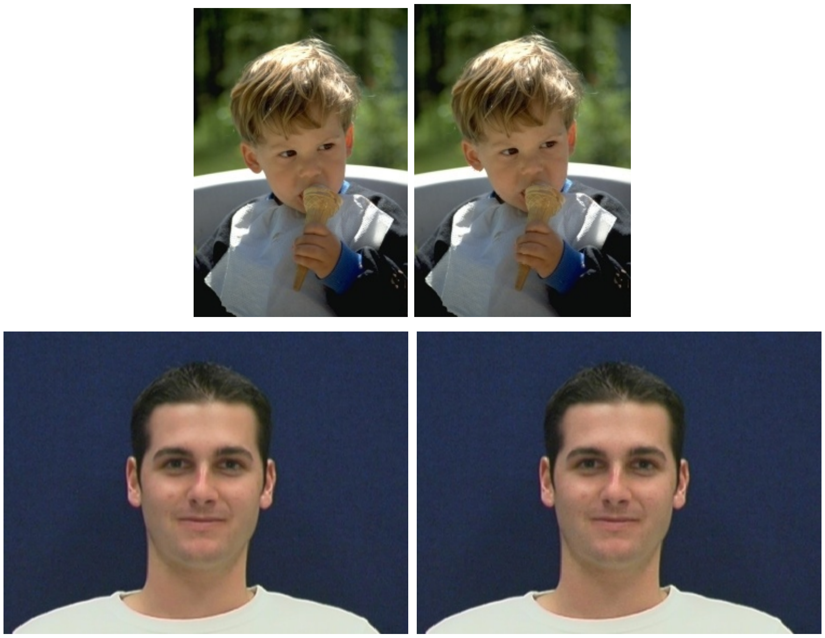


Figure 6.25: Visual distortions, (left column) original images, (right column) stego-images

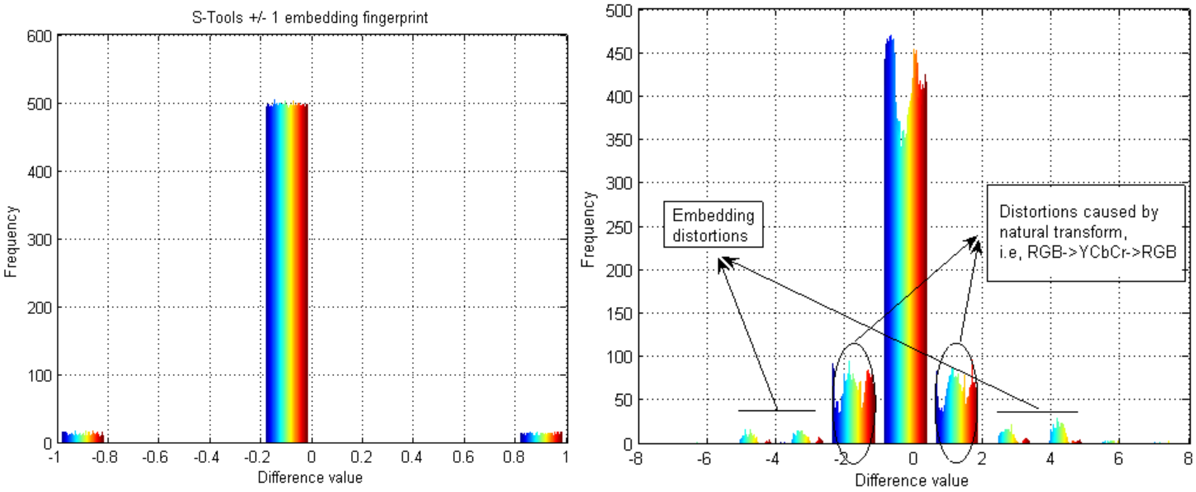


Figure 6.26: Embedding distortion. (Left) the +/- 1 fingerprint of S-Tools and (right) the embedding distortion of the proposed

6.3.3 Limitations and merits

An initial consideration is the limited payload available by targeting skin regions. Extending this method to video files would be a possible remedy. However, a steganographer may choose to consider the entire image for embedding, and then detecting skin area would reduce to just providing the desired secret embedding angle as shown in Figure 6.27 (left) stego wrongly de-rotated to $\theta = -183$ and the retrieved data (right) stego correctly de-rotated to $\theta = -184$ and the retrieved data. See more examples in Appendix F, Figure F.1 and Figure F.2.

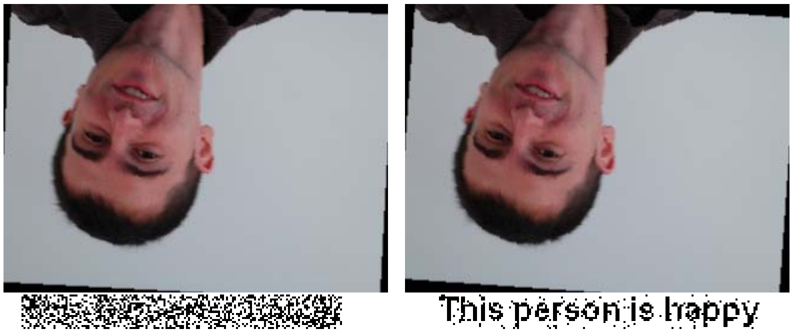


Figure 6.27: Using secret angle, 184°, for the embedding and the extraction

Video-based applications have attracted a lot of attention during the last few years and they are still areas of active research. The proposed *Steganoflage* can be extended to video files. Identifying skin tone regions to embed secret data in videos has the following merits:

- When the embedding is spread on the entire image or frame, scaling, rotation or cropping will result in the destruction of the embedded data as any reference point that can reconstruct the image will be lost. However, skin tone detection in the transformed colour space ensures immunity to geometric transforms.
- The suggested scheme modifies only the regions of the skin tone in the colour transformed channel for imperceptibility.
- The skin-tone has a centre point at C_b, C_r components. It can be modelled and its range is known statistically, therefore, *Steganoflage* can embed safely while preserving these facts.
- If the image, or frame, is tampered with by a cropping process, it is more likely that our selected region will be in the safe zone, because the human faces generally demonstrate the core elements in any given image and thus protected areas, such as portraits.

- The steganographic proposed method is consistent with the object based coding approach followed in MPEG4 and MPEG7 standards, the concept of Video Objects (VOs) and their temporal instances, Video Object Planes (VOPs) is central to MPEG video (Puri & Eleftheriadis, 1998).
- Intra-frame and Inter-frame properties in videos provide a unique environment to deploy a secure mechanism for image based steganography. *Steganoflage* could embed in any frame, e.g., frame #100, an encrypted password and a link to the next frame holding the next portion of the hidden data in the video. Note that this link does not necessarily need to be in a linear fashion, e.g., frames 85→12→3...→n. This can be seen as a macro-scrambling of data.
- Videos are one of the main multimedia files available to the public on the Internet thanks to the giant free web-hosting companies, e.g., YouTube, Google Videos. Every day a mass of these files is uploaded online and human body contents are usually present.

Targeted embedding methods, such as the new enhanced MB2, are faced with much more accurate targeted attacks. That is because "if the selection channel is public, the attacker can focus on areas that were likely modified and use those less likely to have been modified for comparison/calibration purposes" (Kodovsky & Fridrich, 2008a). Nevertheless, the proposed scheme has some advantages. Choosing a specific secret embedding angle would help existing attacked algorithms fool steganalysis tests. Moreover, when an image is de-rotated to its pristine angle state, interpolation occurs, $\theta \notin \{0,90,180,270,360\}$, offering a practical method for minimizing the embedding impact. Identifying skin areas will give an instant direct split of an image into two main areas, one for embedding and another to correct for any statistical distortion caused.

Various authors using wavelet-based steganography face the problem of under/over flow for reconstructed integers that exceed the allowable limits {0, 255}, moreover an additional round-off error problem caused by floating point precision of wavelet coefficients put the embedded bit stream at risk, see examples in (Liu et al., 2006). This is not limited to wavelets but the DCT transform also exhibits round-off and clipping that introduce the possibility of subsequent decoding errors (Miller et al., 2004). To cope with these difficulties, authors such as (El Safy et al., 2009), (Lee et al., 2007) and (Ramani et al., 2007) choose to use the Integer Wavelet Transform, IWT, employing lifting schemes, in lieu of DWT, which maps image intensity integers to integers coefficients and vice versa.

This work, on the other hand, considers, as was discussed in Chapter 4, the DWT for embedding. The $\mathrm{RGB} \rightarrow \mathrm{YC_bC_r}$ provides intensity adjustments automatically and therefore there is no need neither to go for the complication of derivation of conditions as suggested by (Lee et al., 2007) nor to attempt a forced manual adjustments as suggested by (El Safy et al., 2009), which eventually produces severe unnatural distortions to the carrier.

Surmounting the round-off error does not need estimation as advocated by (Lee et al., 2007). All that is needed is the consideration of the integer part in the wavelet coefficients, significant coefficients. Similarly, DCT-based algorithms “contain only a few significant coefficients”, so the capacity is limited, (Shih, 2008, p.121).

The approximation sub-band in the IWT is a sub-sampled copy of the original with values ranging from 0 to 255, on the other hand DWT provides larger values which can accommodate more hidden bits or embed one bit robustly, e.g., 567.500876340983230. These values increase monotonically and compress with further decompositions. IWT extracted hidden bits suffer from error inference, this can be seen from the contrast made in Figure 6.28 and also from the output given in (Ramani et al., 2007).

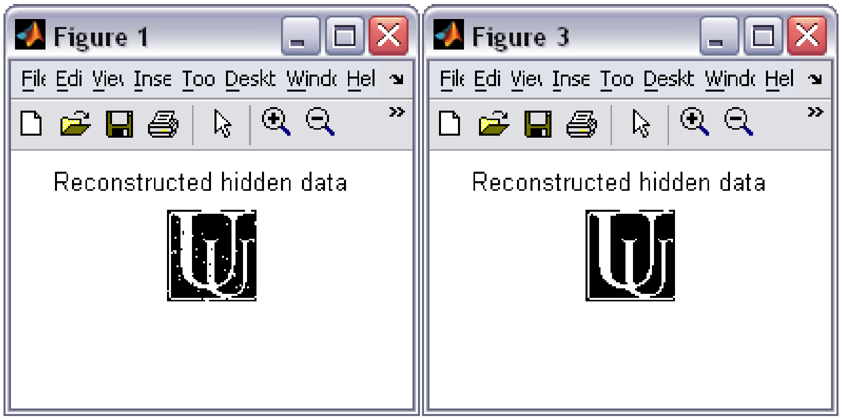


Figure 1
File Edit View Insert Tools Desktop Window Help
Figure 3
File Edit View Insert Tools Desktop Window Help

Reconstructed hidden data

Reconstructed hidden data

Figure 6.28: Extracted hidden data using IWT. (Left) and using proposed (right)

For the sake of comparison and without adhering to the advocated object-based embedding, a test was carried out to examine and measure the distortion caused by utilising the full capacity of the carrier, in our case 0.25 bits/pixel. The comparison to other related methods, using where possible similar images, is tabulated in Table 6.7. The table shows that the proposed algorithm has the least visual distortion to the cover images after embedding 65536 bits which is the full capacity of 0.25 bits/pixel. As can be seen from the table, (Ramani et al., 2007) and (Raja et al., 2008) used non-standard images, however their embedding strategies using IWT with just $^{1}/_{5}$ and ¼, respectively, of the embedding capacity of the proposed method caused more distortions compared to the proposed.

Table 6.7: Distortion comparisons with other methods

Method	*Type of Transform*	*Image*	*PSNR*	*Payload*
(Chang et al., 2003)	DCT	Lena (512x512)	46.3	2000 bits
(Lee et al., 2007)	IWT	Lena (512x512)	48	65536
(Ramani et al., 2007)	IWT	Non-standard	43.593	13460
(Raja et al., 2008)	IWT	Non-standard	42.33	17193
(Gao & Chen, 2008)	DWT, 3rd level	Lena (512x512)	44.74	47447
(El Safy et al., 2009)	IWT	Barbara (512x512)	38	65536
(Chang et al., 2007)	DCT	Lena (512x512)	30.34	36850
(Liu et al., 2007)	DCT	Lena (512x512)	46.272	4096
Proposed	DWT, 1st level	Lena (512x512) Barbara (512x512)	49.8913 50.9024	65536 65536

6.4 Summary

This chapter provides an evaluation of *Steganoflage*. Comparisons with other relevant methods show the advocacy of the introduced *Steganoflage* with its three innovative components, i.e., encryption, skin-tone detection and embedding strategy. The evaluation and analysis of the proposed method point to the different achieved enhancements.

A comparison of each component against related methods has been given. This chapter starts with a comprehensive security analysis test-bed of the proposed image encryption method which includes: key space analysis, key sensitivity analysis, adjacent pixels analysis, randomness test, differential analysis and other security issues. Then an evaluation of the proposed skin tone detection algorithm is highlighted in terms of accuracy and computational complexity. Finally an evaluation of the overall robustness of *Steganoflage* is given which comprises: robustness against intentional and passive attacks, steganalysis, visual perceptibility and the limitations and merits. All of the above evaluations indicate that *Steganoflage* has met its objectives with regard to robustness, security and imperceptibility and shows improvements over current algorithms.

CHAPTER

SEVEN

Conclusion and Future Work

Steganography, the science of secret communication, has received much attention from the scientific community recently. Conferences dedicated to steganography have become more popular and its presence in high impact journals has also increased. This Chapter concludes the thesis by summarizing the research outlined in this thesis, Section 7.1, followed by a discussion on the relation to other work discussed in Chapters 2 and 3 in Section 7.2. Future work is outlined for future investigations in Section 7.3.

7.1 Summary

This thesis has investigated a novel approach to image steganography which provided enhancements to the current available steganography algorithms. The focus was not just on the embedding strategy, as is the trend in recent research, but was also on the pre-processing stages such as payload encryption and embedding area selection.

A comprehensive review of previous work in digital image steganography was discussed and classified into three main categories based on the embedding strategy. The three categories are: spatial domain methods, frequency domain methods and adaptive methods. Advantages and disadvantages of algorithms within each category have been highlighted where possible.

It was observed that all of the current algorithms rely heavily on the conventional encryption systems which for various outlined reasons, highlighted in Chapter 6, do not serve well in the context of image steganography. A second observed fact was that in the search for non-smooth regions for embedding, the introduced algorithms were time consuming and inefficient. These two facts were the drive behind some of the contributions reported in this thesis, namely digital image encryption and skin-tone detection. These contributions were preceded by the relevant literature review pertaining to each method. Security and accuracy tests, concerning the encryption method and skin-tone method respectively, have been inspected. In short the encryption algorithm aimed to extend Secure Hash Algorithm, SHA-2, to encrypt 2D data, while the introduced skin-tone method revived the abandoned luminance which has been proven to be useful in discriminating skin and non-skin areas in a given image.

Exploiting the benefits brought by these two algorithms, a new system named *Steganoflage* was created which used an object-oriented embedding strategy. The results were promising and outperformed relevant methods. *Steganoflage* embeds data using the Reflected Binary Gray Code, RBGC, in the wavelet domain which proved to be robust with less distortion to the carrier file. Security tests were applied to verify the strength of the algorithm including steganalysis based on the 274-dimensional merged vector comprising DCT features and calibrated Markov features.

Applications of *Steganoflage* were considered which were detailed in Chapter 5. The examined applications are: combating digital forgery, multilayer security for patients' data storage and transmission and finally the digital reconstruction of lost signals.

Particularly this thesis highlights the following contributions:

1) A comprehensive state-of-the-art literature review of digital image steganography which was discussed in Chapter 2. The review also comprised critiques, analysis and recommendations.

2) Toward the objective of building up a robust yet flexible steganographic package a second contribution was conceived. Unlike text encryption, image based encryption algorithms remain inefficient due to some inherent properties in bulky data such as digital images. Additionally, encrypted data whose bit sequences comply with

requirements in the steganographic scenario must be met. A balanced bit stream of 1s and 0s and robustness against the avalanche property, i.e., producing poor decryption if some bits are flipped during the transfer, are among those requirements. The introduced encryption algorithm was proven to be efficient and adhered to the aforementioned properties, see Chapter 4 for the algorithm's theoretical formulation and Chapter 6 for analysis and comparison with other methods.

3) Avoiding smooth regions when embedding the secret bits is currently tackled by using an arbitrary window that scans the cover-image in a raster scan fashion and analyzing the texture locally. Normally textural analysis is carried out by calculating the variance, entropy, correlation and other statistical variables. Another way that instantly provides a global area for embedding that exhibits a sufficient textural complexity and is invariant to rotation and translation is through identifying skin-tone regions. The study in Chapter 2 shows that the current skin-tone detection algorithms are either ineffective or time consuming. This put the urgency of providing a real-time and efficient algorithm as a priority. Thus a novel skin-tone detection method was formulated in Chapter 4 and its performance was discussed in Chapter 6.

4) In the course of answering the question “how to embed?” the concept of Object-Oriented embedding (OOE) was investigated. The embedding process took place in the wavelet domain using the reflected binary gray code and guided by the skin-tone area. The integration of these algorithms established an effective property that enabled *Steganoflage* to create a secret angle for embedding. The level of advocacy achieved using *Steganoflage* was satisfactory. Even when omitting the concept of OOE, the unique interaction of the encryption algorithm and the embedding strategy has placed *Steganoflage* ahead of the current methods as seen in Chapter 5 and Chapter 6. Table 7.1 includes relative evaluations of *Steganoflage* performance with that of other steganographic systems.

7.2 Relation to Other Work

The proposed method *Steganoflage* is related to work in the area of watermarking. The objective of the following section is to differentiate *Steganoflage* from these related methods.

A summary of the drawbacks of the current steganographic techniques in the literature and a list of the main characteristics underlying the proposed method of this thesis are summarized in Table 7.1.

Table 7.1: Drawbacks of current steganography methods and benefits of *Steganoflage*

Method	*Descriptions*
Spatial domain techniques	▪ Large payload but often offset the statistical properties of the image ▪ Not robust against lossy compression and image filters ▪ Not robust against rotation, cropping and translation ▪ Not robust against noise ▪ Many work only on the BMP format ▪ Do not address encryption of the payload or use conventional algorithms
DCT domain techniques	▪ Less prone to attacks than the former methods at the expense of capacity ▪ Breach of second order statistics ▪ Breach of DCT coefficients distribution ▪ Work only on the JPEG format ▪ Double compression of the file ▪ Not robust against rotation, cropping and translation ▪ Not robust against noise ▪ Not robust against modification of quantization table, i.e., re-compression ▪ Do not address encryption of the payload or use conventional algorithms
Steganoflage	▪ Object-oriented embedding, OOE ▪ Small embedding space at the benefit of robustness. Resolved by targeting video files ▪ Resistance to rotation, translation, cropping and moderate noise impulses ▪ No known statistical vulnerabilities ▪ Resistance to lossy compression thanks to the DWT ▪ Performs better than DCT algorithms in keeping the carrier distortion to the minimum ▪ Addresses a novel encryption method of the payload

Additionally, spatial domain approaches are vulnerable to attacks for the following reasons, not exhaustive however:

- spatial domain techniques provide only a spatial description for an image at the pixel level, i.e., {0 255} for 8-bit image files
- they can easily be fooled by any linear or non linear distortion of the image, hence they cannot tolerate compression or noise
- since colour components of an RGB image are highly correlated, embedding in the spatial domain distorts the natural statistical properties of an image file more than that in the frequency domain which leaves spatial domain methods exposed to attacks.

7.2.1 Region-based image watermarking

The idea of embedding into particular regions has been previously articulated by Nikolaidis and Pitas (Nikolaidis & Pitas, 2000) and (Nikolaidis & Pitas, 2001) who focused mainly on watermarking. Their two main publications involved a method for handling colour images (Nikolaidis & Pitas, 2000) and another method for greyscale images (Nikolaidis & Pitas, 2001): In the earlier work, Nikolaidis and Pitas introduced a method for embedding and detecting a chaotic signature, watermark, in the spatial domain of colour facial images by localizing facial features. The features' role was to label an area in which the watermark was embedded and detected. The Renyi map along with Peano scanning were used to generate the chaotic signature which was embedded in a facial region segmented using the HSV colour space. In the second publication, Nikolaidis and Pitas used the classical K-means algorithm to segment greyscale images and then voted for the best fit elliptical shape blobs whose bounding rectangles were chosen as the embedding area for the watermark. Table 7.2 contrasts the proposed method *Steganoflage* with the related work of Nikolaidis and Pitas.

Table 7.2: Comparision of *Steganoflage* against Nikolaidis and Pitas' work

Criterion	*Steganoflage*	(Nikolaidis & Pitas, 2001)	(Nikolaidis & Pitas, 2000)
Applicability	Steganography	Watermarking	Watermarking
Objective	Secret communication	Copyright preservation	Copyright preservation
Encryption	A new method	Renyi map	Renyi map
Skin-tone detection	A new method	N/A	HSV-based
Greyscale image segmentation	A new method , Voronoi diagram	K-means algorithm	N/A
Decoding	Exact extraction	Detection of presence by cross-correlation	Detection of presence by cross-correlation
Domain	Wavelet domain	Spatial domain	Spatial domain

7.2.2 Self-embedding

Luo (Luo et al., 2008) proposed a self-embedding pixel-wise and block-wise algorithm that took the advantage of digital half-toning. A copy of the image itself was embedded into the LSB of the image in the spatial domain for tamper detection. The method was considered fragile as the hidden bits in the LSB were easily attacked by intentional alterations or common image operations (Luo et al., 2008, p.168).

Half-toning methods, due to their compactness, are very sensitive and therefore suffer even when legitimate or unintentional alterations occur, e.g., JPEG compression, interference of noise. Hence, protecting this sensitivity dictates migrating to the frequency domain as in *Steganoflage*.

7.3 Future Work

Today's world of digital media is in a constant state of evolution. Steganography is regarded as technology that has major competitive applications. In this regards, future work is set mainly to increase the robustness against digital-analogue-digital distortions which is also known as the print-scan resilience. Additionally, improving the algorithm to be able to withstand severe JPEG/MPEG compression would be a challenge.

7.3.1 Resilience to print-scan distortions (secure ID card)

Steganoflage could be used as an innovative security solution to counteract innovative criminals via preventing the forgery of important personal documents, e.g., identity theft. Individuals passports, ID cards and driving licenses are among the documents that fraud criminals are after. In July 2005, 2nd WWII files at the National Archives, UK, were found to contain forged documents. Forensics experts stated that the forgery took place during or after the year 2000.

Identification cards are prone to forgery in aspects relating to biodata alteration or photo replacement. To protect photos, government bodies use a physical watermark on the photos using a steel stamp which is half visible or sometimes they use a rubber stamp. Systems on chip, on the other hand, are extremely expensive to roll out and require dedicated hardware and some chip circuits can be reverse engineered.

Using steganography to embed the document’s biodata into itself can provide a secure and viable authentication. Moreover, since it is a software based tool the cost associated with the development will be very low. This way tampering becomes highly challenging because the embedded data is inseparable from the document. Little progress in the literature has been made owing to the complexity of the problem (Solanki et al., 2006).

7.3.2 Resilience to severe image lossy compression (iPhone)

Current technology stimulates the subtle deployment of steganography into portable devices such as the iPhone. Hence, the proposed algorithm needs to be revisited in order to improve its functionality with a customised outlook that fits such devices and their bandwidth. To this end, enhancements against severe image compression are necessary.

7.3.3 Tamperproof CCTV surveillance

Driven by the inexpensive means of data archiving, the ease of manipulation and transmission, some surveillance recording devices have gone digital. Until now, there has been no system to detect the unauthorised manipulation of such video footages apart from basic encryption. The ease of editing visual data in the digital domain has facilitated unauthorized tampering performed without leaving any perceptible traces. Therefore recorded CCTV, Closed-Circuit TeleVision, video does not stand up in court as a 100% reliable evidence. Most of the work done so far on CCTV surveillance dealt with object detection, object recognition, tracking, behaviour analysis and image retrieval.

Steganoflage can be extended for frames' self-embedding and also to embed additional bytes, metadata, which could be useful for query purposes such as: unique reference, date and time stamp, officer name, officer number, location, operation detail. Consequently, the algorithm proposed in this thesis must run in real-time. The time complexity of the embedding stage needs a speed enhancement. Figure 7.1 displays the core idea of this solution.

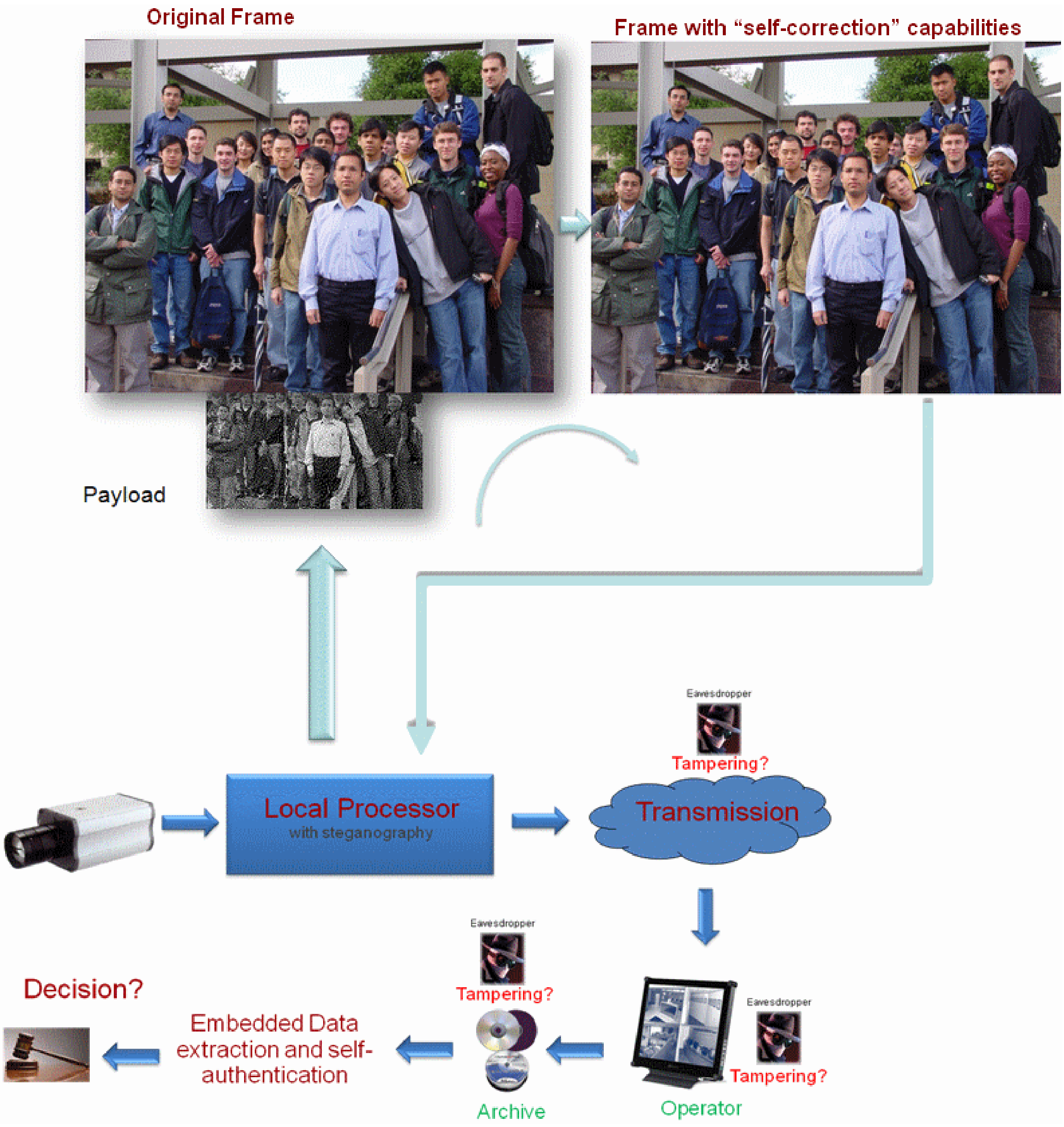


Figure 7.1: Simplified theoretical framework of a tamperproof surveillance system

7.4 Conclusion

The objective of this research was to enhance steganography in digital images. Hence a new approach was developed, *Steganoflage*, which implemented an object-oriented embedding. *Steganoflage* constituted a unique architecture with efficient components, namely, image encryption, skin-tone detection and wavelet embedding using the RBGC coding. Evaluations of each component provided evidence to confirm the hypotheses of this research. The outcome of these evaluations highlighted the potential of *Steganoflage* in improving upon existing methods. Tables of comparison were developed throughout the thesis to compare the performance of each component against related approaches. Furthermore, tables were constructed to compare *Steganoflage* against associated methods. Future work could focus on the print-scan resilience issue which can augment the capabilities of *Steganoflage*. Surmounting severe compression would add value to the method meeting the requirement for the wireless transmission technology. Another interesting avenue for future work could be a tamper proof CCTV surveillance system.

Finally, the introduced object-oriented embedding with the new lightweight encryption algorithm and the real-time and efficient skin-tone detection method are unique to *Steganoflage*, however these independent components could be potentially deployed in a multitude of multimedia application domains. To sum up, the overall thesis contribution lies in the development of a new steganographic approach which is shown to be both robust and imperceptible and with this in mind *Steganoflage* is developed.

Appendices

Appendix A: Bridging MATLAB to a Web Scripting Language

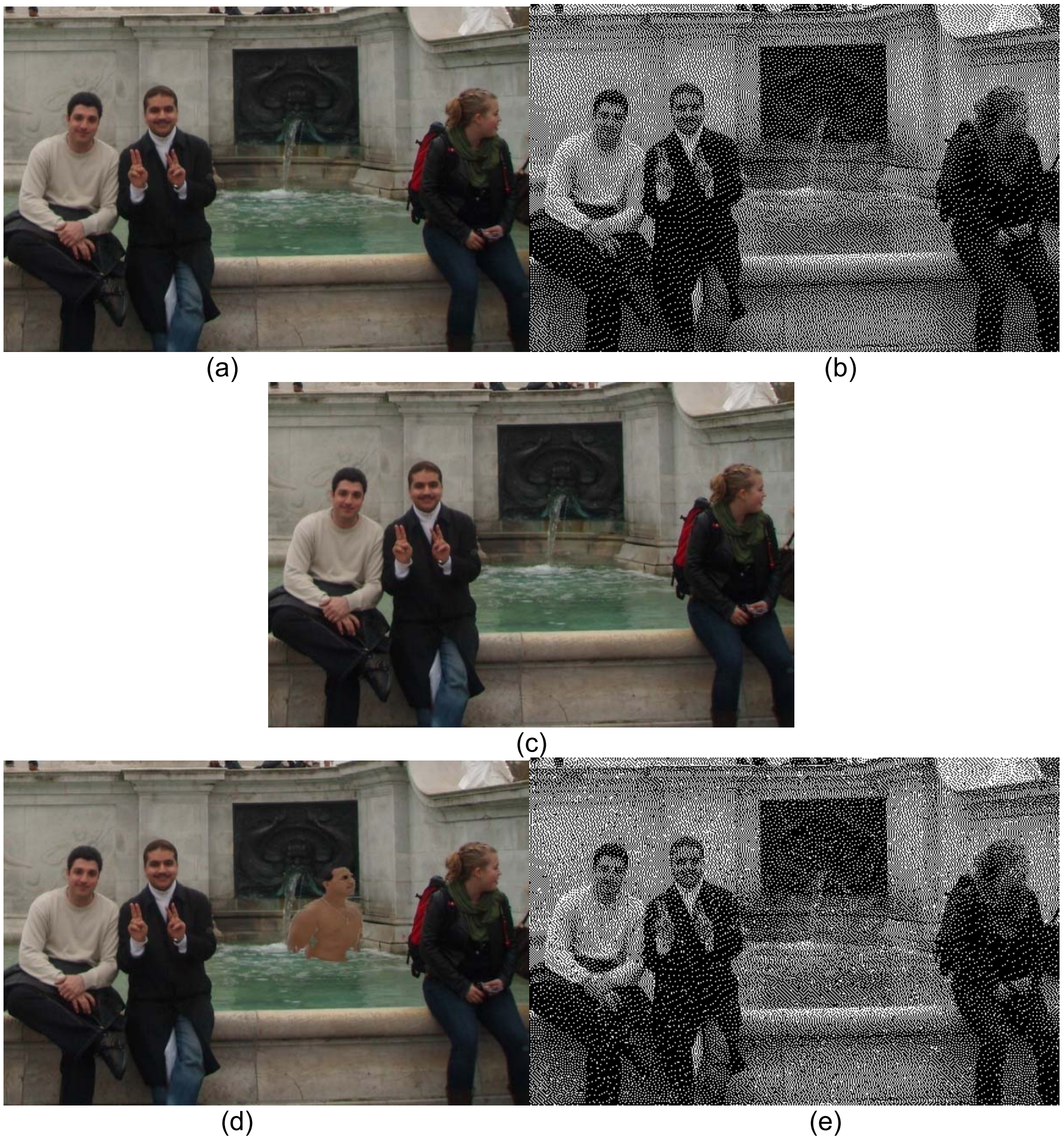
The following code describes the implementation of Web scripts, e.g., HTML, JavaScript, PHP, into MATLAB commands. This code writes to an external file, usually .HTML or .HTM to enable viewing of the output on the browser.

```
Steg=Code_Gray(Cover,Secret,PassI,PassII);

% The main function is called here. The parameter list comprises:
% Cover image ‘string’  Secret image ‘string’   Password hash functions
%‘string’  PassI=SHA2(‘Pass’) and PassII=SHA2 (PassI)
imwrite(Steg,sprintf('Steg_%s',Steg));
copyfile(Cover, 'C:\wamp\www\UPLOAD\')
fid = fopen('Report.html','w');

% opens the file Report.html for writing or creates it if necessary
f=clock;
% starts the clock
time_create=sprintf('%02.0f:%02.0f:%02.0f', f(4),f(5),f(6));
fprintf(fid,'<body background=\"back.jpg\"><center><font color=navy>This file
was created via SteganoFlage''s web interface on  %s  at:  %s
</font>',date,num2str(time_create));
fprintf(fid,'<br><hr>');
fprintf(fid,'<br>');
fprintf(fid,'<font FACE="ARIAL" color=brown><H2>SteganoFlage Generated Report
</H2><font color=brown><center>Faculty of Computing and Engineering
<br>University of Ulster at Magee <br> Londonderry, BT48
7JL.</center></font></b></font>');
fprintf(fid,'<br>');
fprintf(fid,'<hr></center>');
fprintf(fid,'<div align=center>');
fprintf(fid,'<table border=0 width=100%%>');
fprintf(fid,'<tr><td>Your stego image: <br><img src=\"Steg_%s\"></td>',Cov);
fprintf(fid,'<td>Your input image: <br><img src=\"%s\">',Cov);

% fprintf function writes HTML code and page layout to the external HTML file.
% Certain reserved characters in MATLAB can escape evaluation
% if preceded with ‘\’ as in =\"back.jpg\"
```

Figure A.1: Parsing HTML code into MATLAB commands

Appendix C: Self-Embedding Examples

The following figures depict one of the applications where *Steganoflage* was used to provide tamper-proof digital files by self-embedding in the wavelet domain for authentication.



(a) (b)

(c)

(d) (e)

Figure C.1: Doctored image-Victoria Memorial: (a) the original image (b) half-tone copy (c) self-embedded image (d) tampered *c* and (e) the recovered copy

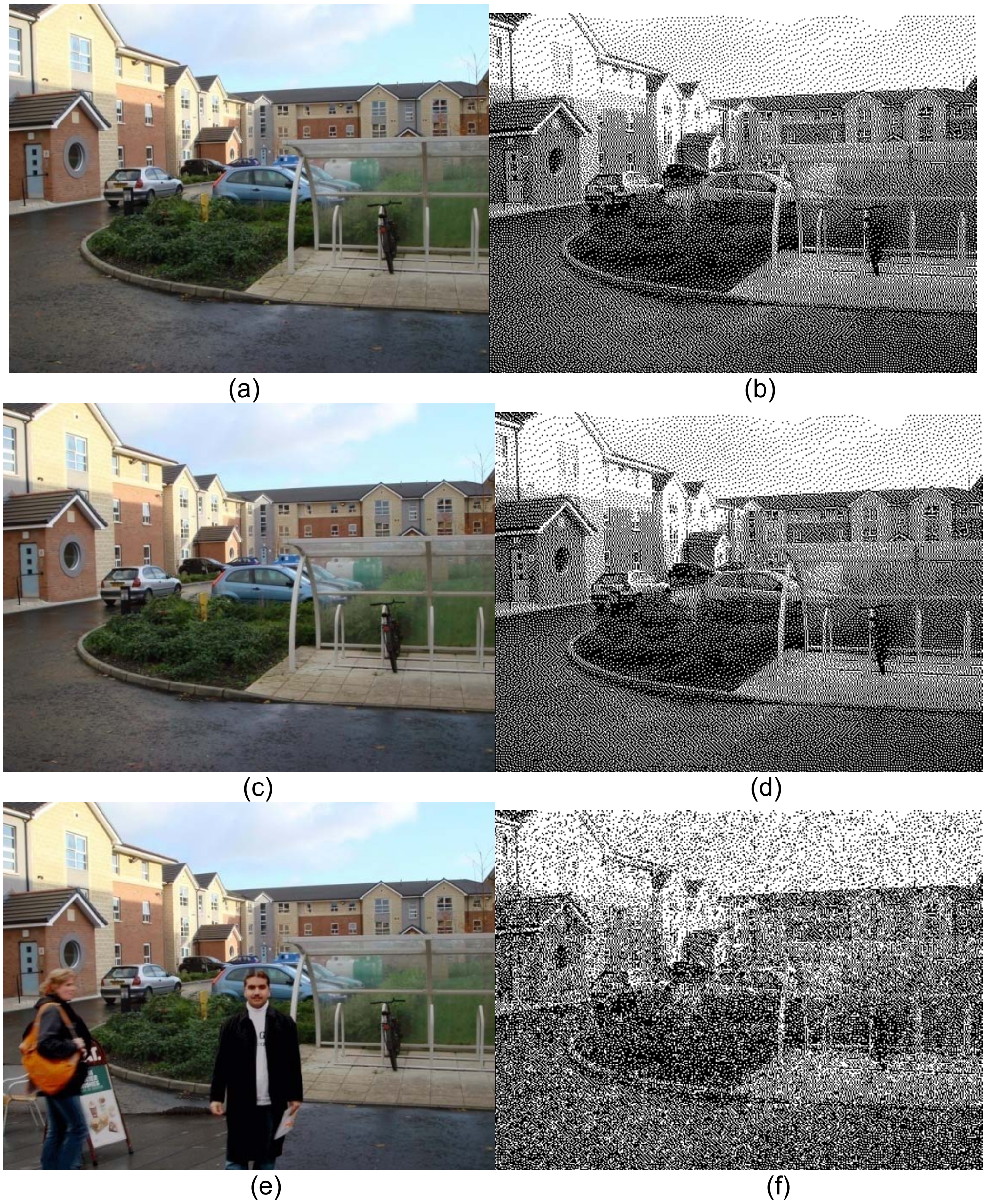
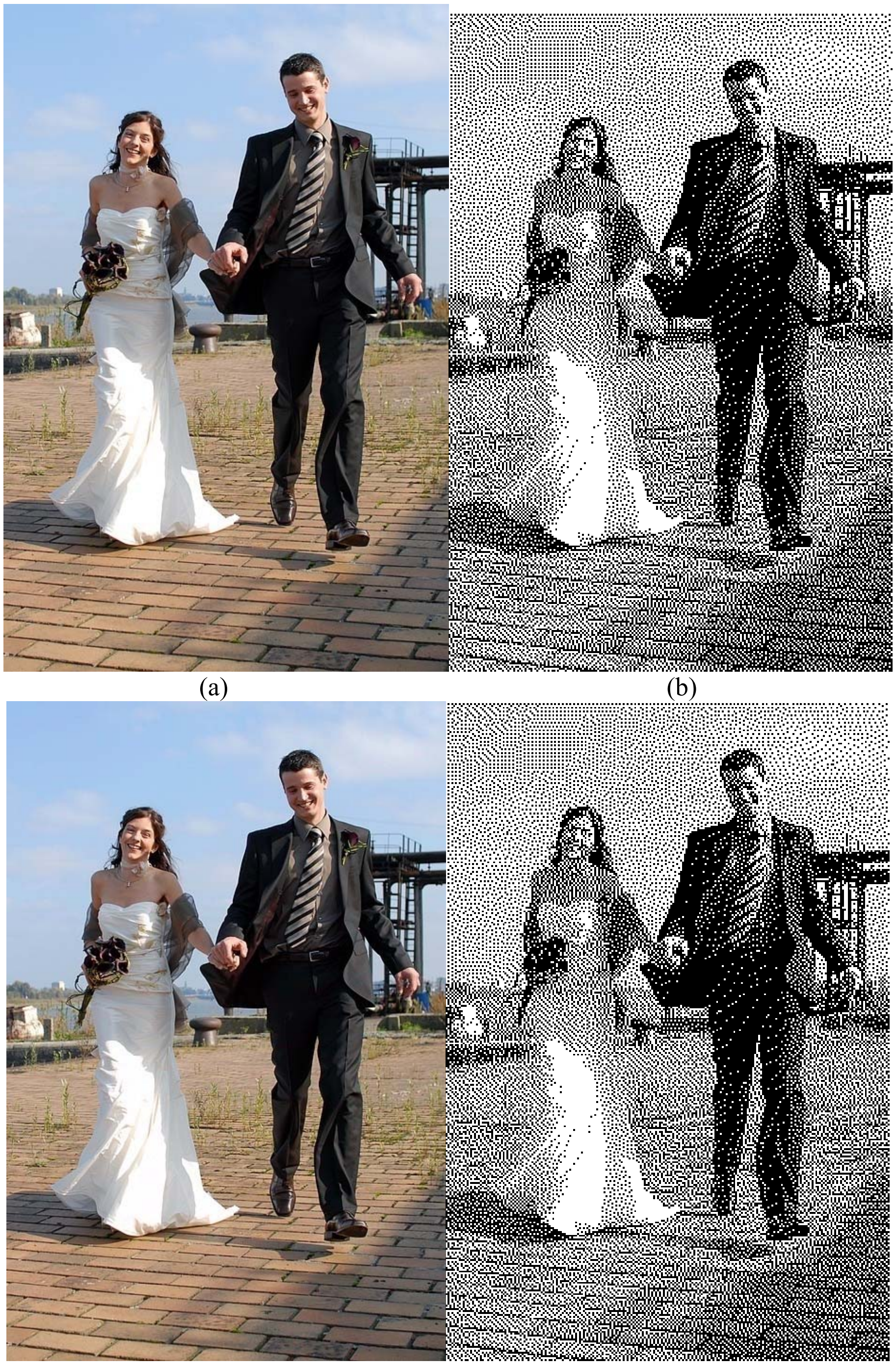


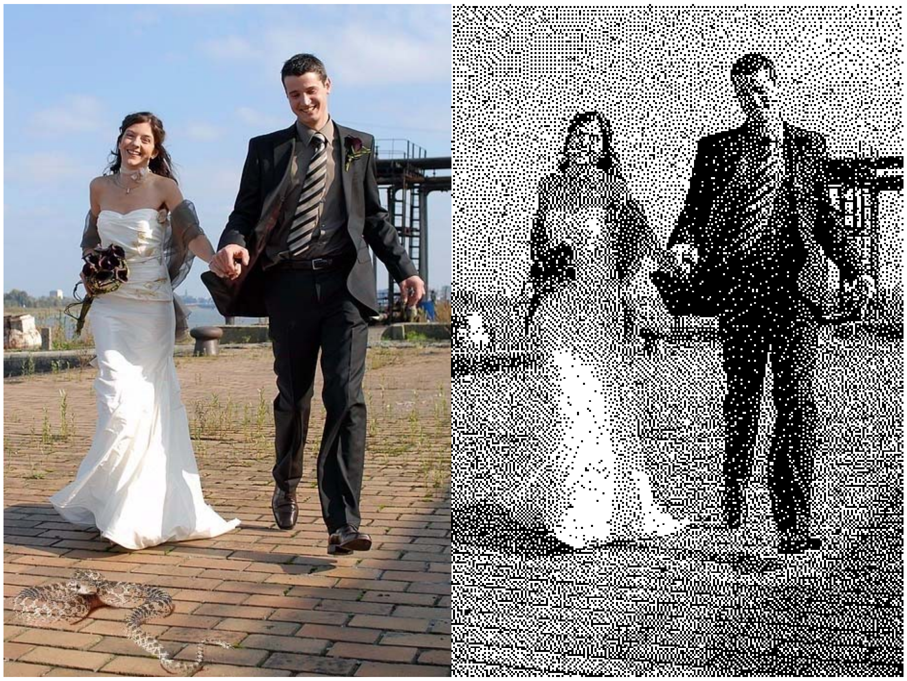
Figure C.2: Doctored image-Duncreggan student village: (a) the original image, (b) half-tone copy, (c) self-embedded image, (d) extracted copy without any attack, (e) tampered *c* and (f) the recovered copy after attack



(a) (b)

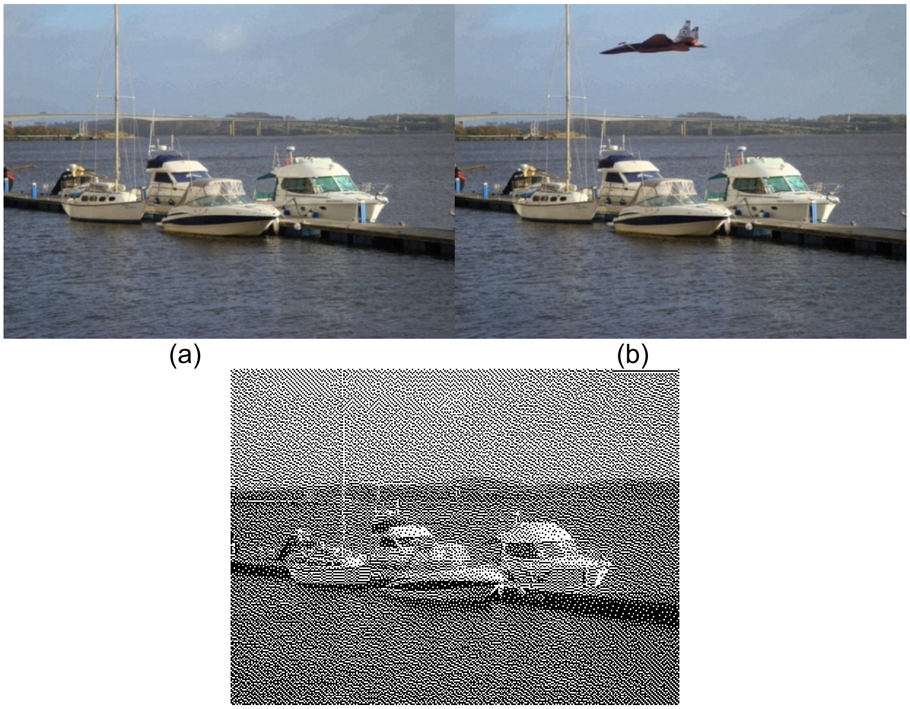
(c) (d)

Figure C.3: Doctored image-couple: (a) the original image, (b) half-tone copy, (c) self-embedded image, (d) extracted copy without any attack, (e) tampered *c* and (f) the recovered copy after attack



(e) (f)

Figure C.3: (Cont…)



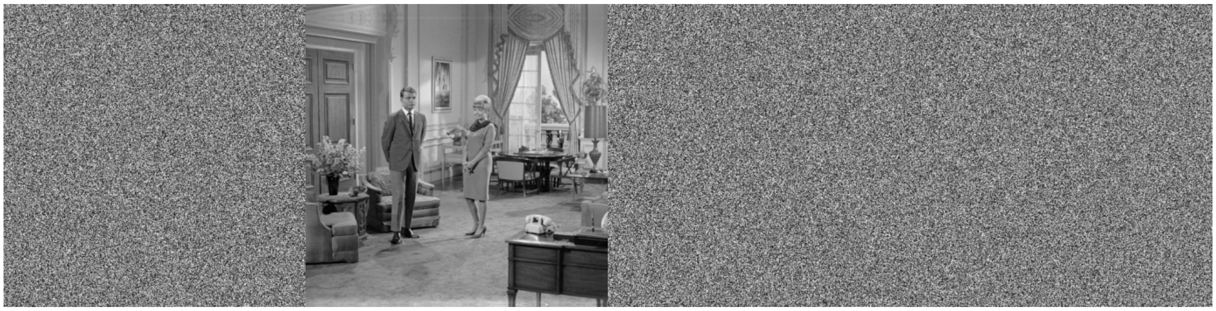
(a) (b)

(c)

Figure C.4: Doctored image- River Foyle: (a) the self-embedded image, (b) tampered *a* and (c) the recovered copy after attack

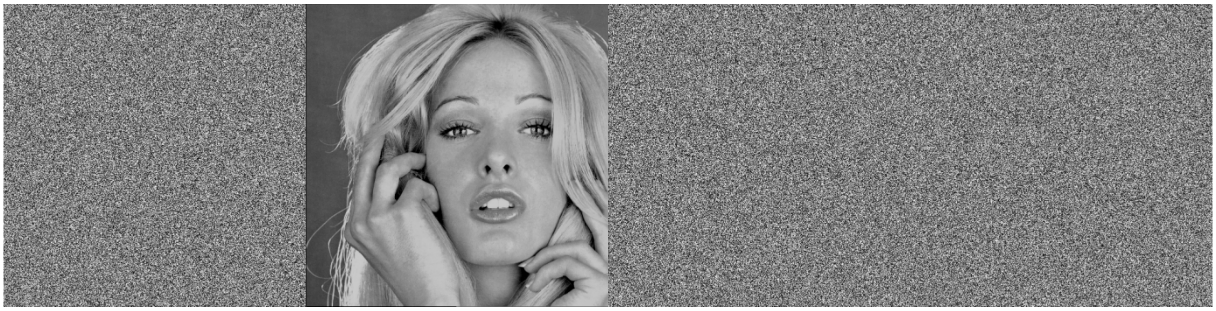
Appendix D: Key Sensitivity Analysis of the Image Encryption

The following figures provide additional experiments that complement the ones shown in Section 6.1.2. This re-affirms that *Steganoflage* displays high sensitivity to the key initial conditions.



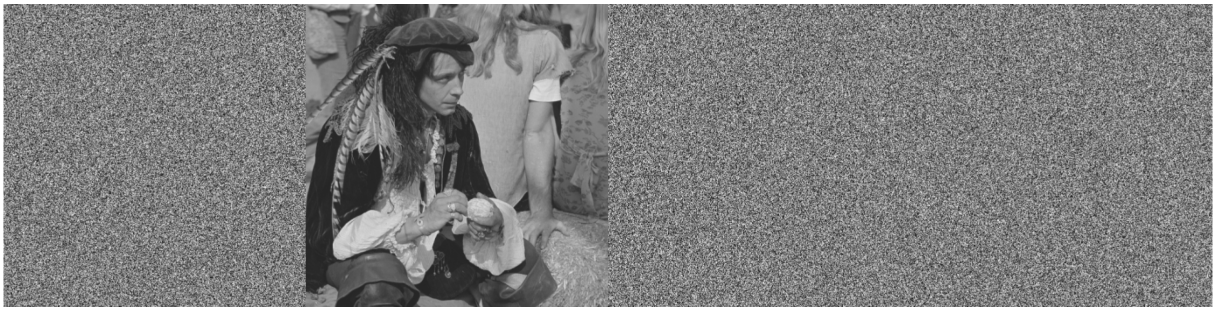
(a) (b) (c) (d)

Figure D.1: Test-1: (a) the encrypted image; (b) the decrypted image (a) using the correct key ‘04082009’ '3b958af0fdc44c56f16**0**bd2e044c26ee461cd2d4'; (c) the decrypted image (a) using the wrong key ‘03082009’ '23df6f849d7f60d50c9a33715497b473d79e34dc'; (d) the decrypted image *(a)* using a slightly modified hash ‘3b958af0fdc44c56f16**1**bd2e044c26ee461cd2d4’



(a) (b) (c) (d)

Figure D.2: Test-2: (a) the encrypted image; (b) the decrypted image (a) using the correct key ‘Abbas’ '957c7476bbb5830985ebe51302382d5**8**520788f6'; (c) the decrypted image (a) using the wrong key ‘abbas’ '592a598416630f00be6d84815f03eaa2378346bc'; (d) the decrypted image *(a)* using a slightly modified hash ‘957c7476bbb5830985ebe51302382d5**9**520788f6’



(a) (b) (c) (d)

Figure D.3: Test-3: (a) the encrypted image; (b) the decrypted image (a) using the correct key ‘Encryption’ ' 0af**1**49c2ed169b89f192451f70ee9a3ae70eab02'; (c) the decrypted image (a) using the wrong key ‘EncrYption’ ' 69ff5a14d81d310068783d1f1872fea7ba0128d5'; (d) the decrypted image *(a)* using a slightly modified hash ‘0af**0**49c2ed169b89f192451f70ee9a3ae70eab02’

Appendix E: Dark Skin-Tone Detection

Additional experiments for skin-tone detection targeting African look people that supports Section 6.2 is shown below. *Steganoflage*'s skin-tone algorithm is insensitive and has a uniform performance across all races.

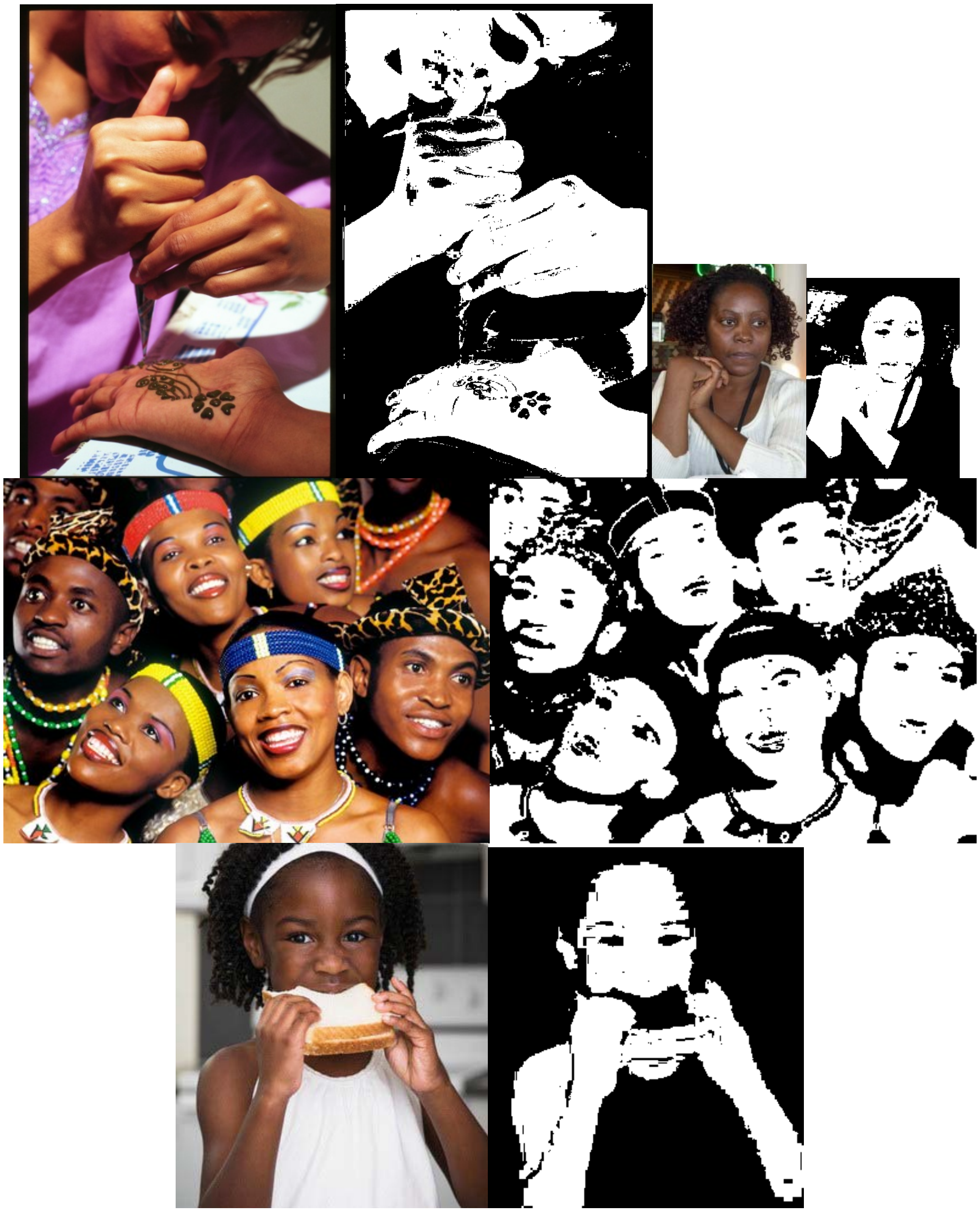


Figure E.1: The proposed skin-tone detection algorithm performance on dark skin

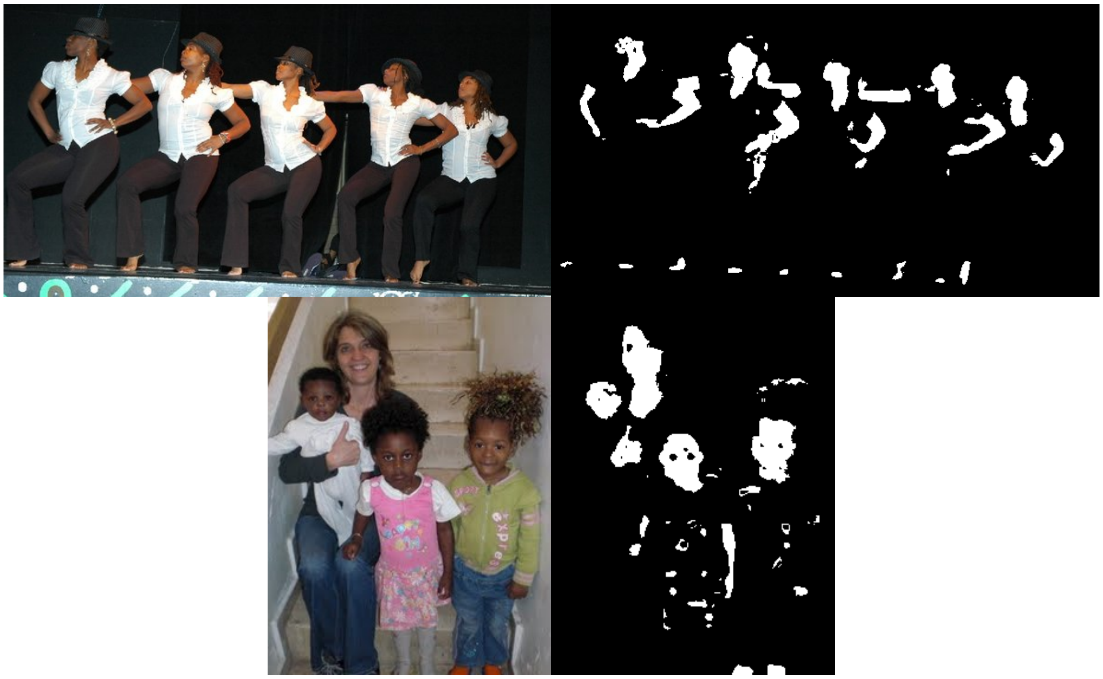


Figure E.1: (Cont…)

Appendix F: Lossy Embedding with a Secret Angle

The following figures show *Steganoflage*'s unique embedding strategy using two secret keys, one to encrypt the data and another to provide the right angle for embedding. Without which the extraction of the embedded data is deemed impossible.

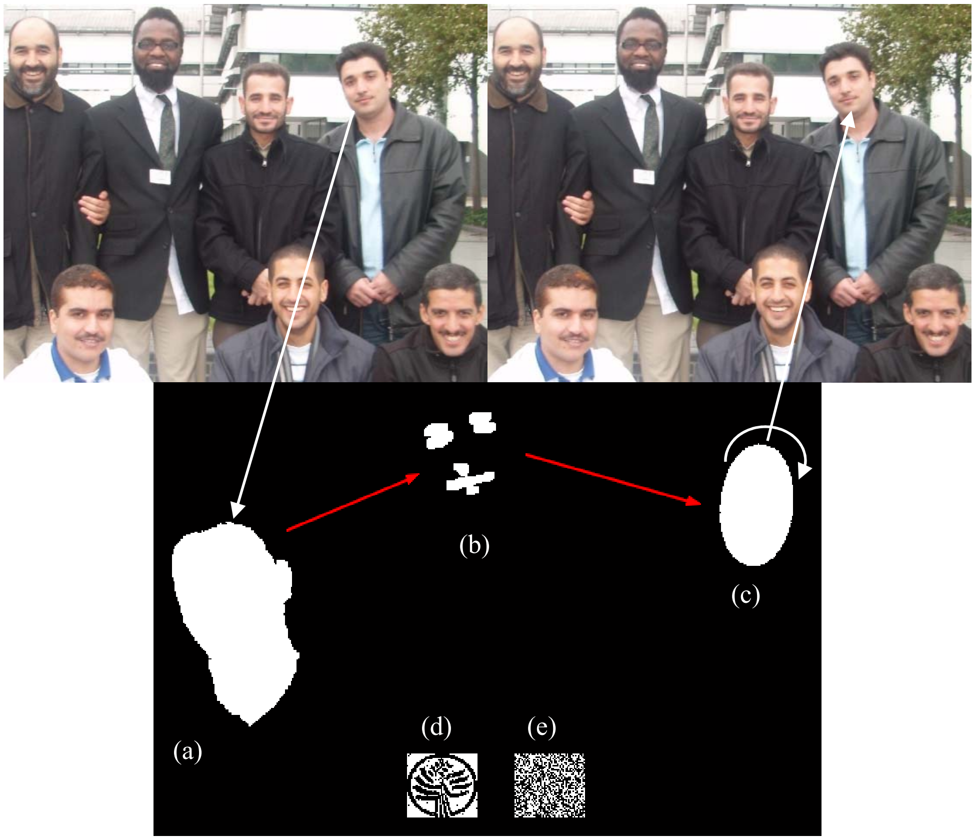


Figure F.1: Secret angle embedding- Example I: (top-left) original image, (top-right) stego-image, (bottom) face detection (a), feature extraction (b), ellipse constructed using eyes location and rotated with theta=-10^o (c), payload extracted using the correct angle theta=-10^o (d) and using the wrong angle theta=-9^o (e)

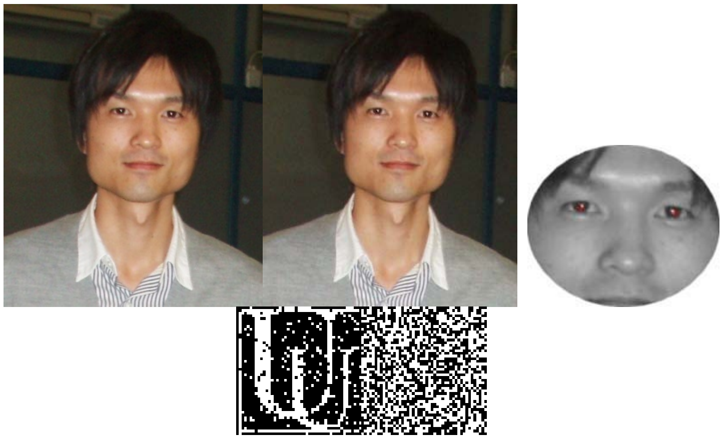


Figure F.2: Secret angle embedding- Example II: (top left to right) original image, stego-image and eyes location, (red stars), respectively, (bottom left to right) payload extracted using the correct angle theta=-28° and using the wrong angle theta=-27°, respectively

References

Abadpour, A. & Kasaei, S., 2005. Pixel-based skin detection for pornography filtering. *Iranian Journal of Electrical and Electronic Engineering*, 1(3), pp.21-41.

Abdelwahab, A.A. & Hassan, L.A., 2008. A discrete Wavelet transform based technique for image data hiding. In *Proceedings of 25th National Radio Science Conference, NRSC 2008*. Egypt, 2008. March 18-20.pp.1-9.

Abdulaziz, N.K. & Pang, K.K., 2000. Robust data hiding for images. In *Proceedings of IEEE International Conference on Communication Technology, WCC-ICCT*. Beijing , China, 2000. 21-25 Aug. pp. 380-383.

Abdullah-Al-Wadud, M. & Chae, O., 2008. Skin segmentation using color distance map and water-flow property. In *Proceedings of International Conference on Information Assurance and Security*. Italy, 2008. 8-10 Sept. pp. 83-88.

Albiol, A., Torres, L. & Delp, E.J., 2001. Optimum color spaces for skin detection. In *Proceedings of the IEEE International Conference on Image Processing*. Greece, 2001. pp. 122-124.

Alvarez, P., 2004. Using extended file information (EXIF) file headers in digital evidence analysis. *International Journal of Digital Evidence*, 2(3), pp.1-5.

Anderson, R.J. & Petitcolas, F.A.P., 1998. On the limits of steganography. *IEEE Journal of Selected Areas in Communications*, 16(4), pp.474-81.

Areepongsa, S., Kaewkamnerd, N., Syed, Y.F. & Rao, K.R., 2000. Exploring on steganography for low bit rate wavelet based coder in image retrieval system. In *Proceedings of IEEE TENCON*. Kuala Lumpur, Malaysia, 2000.

Ashtiyani, M., Birgani, P.M. & Hosseini, H.M., 2008. Chaos-based medical image encryption using symmetric cryptography. In *Proceedings of IEEE 3rd International Conference on Information and Communication Technologies: From Theory to Applications (ICTTA 2008).*, 2008. 7-11 April. pp.1-5.

Avcibas, I., Memon, N. & Sankur, B., 2002. Image steganalysis with binary similarity measures. In *Proceedings of the IEEE International Conference on Image Processing.*, 2002. 24-28 June. pp.645-648.

Bailey, K. & Curran, K., 2006. An evaluation of image based steganography methods. *Multimedia Tools and Applications*, 30(1), pp.55-88.

Bas, P., 2003. *Analyse stéganographique d'images numériques: Comparaison de différentes méthodes*. France: University of Joseph Fourier. 23rd June.

BBC News, 2007. *Hiding messages in plain sight*. [Online] Available at: HYPERLINK "http://news.bbc.co.uk/go/pr/fr/-/1/hi/technology/6361891.stm" http://news.bbc.co.uk/go/pr/fr/-/1/hi/technology/6361891.stm [Accessed 08 September 2009].

Bender, W. et al., 2000. Applications for data hiding. *IBM Systems Journal*, 39(3&4), pp.547-68.

Beniak, M., Pavlovicova, J. & Oravec, M., 2008. Automatic face detection based on chrominance components analysis systems. In *Proceedings of International Conference on Systems, Signals and Image Processing*. Slovak Republic, 2008. 25-28 June. pp. 475-478.

Berens, J. & Finlayson, G.D., 2000. Log-opponent chromaticity coding of colour space. In *Proceedings of IEEE International Conference on Pattern Recognition*. Barcelona, Spain, 2000. pp. 206-211.

Beşdok, E., 2005. Hiding information in multispectral spatial images. *International Journal of Electronics and Communications*, 59(1), pp.15-24.

Bing, L. & Jia-wei, X., 2005. Period of Arnold transformation and its application in image scrambling. *Journal of Central South University of Technology*, 12(1), pp.278-82.

Bogomjakov, A., Gotsman, C. & Isenburg, M., 2008. Distortion-free steganography for polygon meshes. In *Proceedings of Computer Graphics Forum, Eurographics'08.*, 2008. April.

Böhme, R. & Westfeld, A., 2004. Breaking cauchy model-based JPEG steganography with first order statistics. In *Proceedings of the European Symposium on Research in Computer Security, ESORICS 2004*. Valbonne, France, 2004. LNCS. 13th Sept.

Böhme, R. & Westfeld, A., 2005. Exploiting preserved statistics for steganalysis. In *Proceedings of 6th International Workshop Information Hiding*. Toronto, Canada, 2005. Lecture Notes in Computer Science, Springer. 23-25 May. pp.82-96.

Bourdon, R., 2009. *WampServer*. [Online] Available at: HYPERLINK "http://www.wampserver.com/en/" http://www.wampserver.com/en/ [Accessed 04 August 2009].

Buchholz, J.J., 2001. *Matlab Implementation of the Advanced Encryption Standard*. http://buchholz.hs-bremen.de/aes/aes.htm.

Cancelli, G., Doërr, G.J., Barni, M. & Cox, I.J., 2008. A comparative study of +/-1 steganalyzers. In *Proceedings of IEEE 10th Workshop on Multimedia Signal Processing*. Queensland, Australia, 2008. 8-10 Oct. pp.791-796.

Chang, C.C., Chen, T.S. & Chung, L.Z., 2002. A steganographic method based upon JPEG and quantization table modification. *Information Sciences*, 141(1-2), pp.123-38.

Chang, C.C., Chen, T.S. & Hsia, H.C., 2003. An effective image steganographic scheme based on wavelet transformation and pattern-based modification. In *International Conference on Computer Networks and Mobile Computing (ICCNMC'03)*. Shanghai, China, 2003. pp.450-453.

Chang, C.C., Hu, Y.S. & Lu, T.C., 2006a. A watermarking-based image ownership and tampering authentication scheme. *Pattern Recognition Letters*, 27(5), pp.439-46.

Chang, C.C., Lin, C.C., Tseng, C.S. & Tai, W.L., 2007. Reversible hiding in DCT-based compressed images. *Information Sciences*, 177(13), pp.2768-86.

Chang, C.C., Lin, C.Y. & Wang, Y.Z., 2006b. New image steganographic methods using run-length Approach. *Information Sciences*, 176(22), pp.3393-408.

Chang, C.C., Tsai, P. & Lin, M.H., 2004. An adaptive steganography for index-based images using codeword grouping. *Advances in Multimedia Information Processing-PCM*, 3333, pp.731-38.

Chang, C.C. & Tseng, H.W., 2004. A steganographic method for digital images using side match. *Pattern Recognition Letters*, 25(12), pp.1431-37.

Chao, M.W., Lin, C.H., Yu, C.W. & Lee, T.Y., 2009. A high capacity 3D steganography algorithm. *IEEE Transactions on Visualization and Computer Graphics*, 15(2), pp.274-84.

Cheddad, A., Condell, J., Curran, K. & Mc Kevitt, P., 2008a. *An encryption method*. United Kingdom Patent Application No. 0819976.2. University of Ulster.

Cheddad, A., Condell, J., Curran, K. & Mc Kevitt, P., 2008c. *Method for skin tone detection*. United Kingdom Patent Application No. 0819982.0. University of Ulster.

Cheddad, A., Condell, J., Curran, K. & Mc Kevitt, P., 2008d. Securing information content using new encryption method and steganography. In *Proceedings of the 3rd IEEE International Conference on Digital Information Management*. London, UK, 2008d. 13-16 Nov. pp. 563-568.

Cheddad, A., Condell, J., Curran, K. & Mc Kevitt, P., 2008e. Skin tone based steganography in video files exploiting the YCbCr colour space. In *Proceedings of the IEEE International Conference on Multimedia and Expo, ICME*. Hannover, Germany, 2008e. June 23-26. pp. 905-909.

Cheddad, A., Condell, J., Curran, K. & Mc Kevitt, P., 2009a. A new colour space for skin tone detection. In *IEEE International Conference on Image Processing, ICIP 2009*. Cairo, Egypt, 2009a. IEEE Signal Processing Society. 7-11 November.

Cheddad, A., Condell, J., Curran, K. & Mc Kevitt, P., 2009b. A secure and improved self-embedding algorithm to combat digital document forgery. *Signal Processing*, 89(12), pp.2324-32.

Cheddad, A., Condell, J., Curran, K. & Mc Kevitt, P., 2009c. A skin tone detection algorithm for an adaptive approach to steganography. *Signal Processing*, 89(12), pp.2465-78.

Cheddad, A., Condell, J., Curran, K. & Mc Kevitt, P., 2010. Digital image steganography: survey and analysis of current methods. *Signal Processing*, 90(3), pp.727-52.

Cheddad, A., Mohamad, D. & Abd Manaf, A., 2008b. Exploiting Voronoi diagram properties in face segmentation and features extraction. *Pattern Recognition*, 41(12), pp.3842-59.

Chen, W.Y., 2007. Color image steganography scheme using set partitioning in hierarchical trees coding, digital Fourier transform and adaptive phase modulation. *Applied Mathematics and Computation*, 185(1), pp.432-48.

Cheng, J. & Kot, A.C., 2009. Steganalysis of halftone image using inverse halftoning. *Signal Processing*, 89(6), pp.1000-10.

Chen, H.Y., Huang, C.L. & Fu, C.M., 2008. Hybrid-boost learning for multi-pose face detection and facial expression recognition. *Pattern Recognition*, 41(3), pp.1173-85.

Chen, C. & Shi, Y.Q., 2008. JPEG image steganalysis utilizing both intrablock and interblock correlations. In *Proceedings of IEEE International Symposium on Circuits and Systems, ISCAS 2008*. Seattle, Washington, USA, 2008. 18-21 March.

Chen, Y.S. & Wang, R.Z., 2009. steganalysis of reversible contrast mapping watermarking. *IEEE Signal Processing Letters*, 16(2), pp.125-28.

Chen, L.S. & Zheng, G.X., 2005. Chaos-based encryption for digital images and videos. In Furht, B. & Kirovski, D. *Multimedia Security Handbook*. CRC Press. p.133–167.

Chi, M.C., Jhu, J.A. & Chen, M.J., 2006. H.263+ region-of-interest video coding with efficient skin-color extraction. In *Proceedings of International Conference on Consumer Electronics*. Las Vegas, USA, 2006. 7-11 Jan. pp. 381-382.

Choudhury, A., Rogers, M., Gillam, B. & Watson, K., 2008. A novel skin tone detection algorithm for contraband image analysis. In *Proceedings of International Workshop on Systematic Approaches to Digital Forensic Engineering*. California, USA, 2008. 22 May. pp.3-9.

Civicioglu, P., Alci, M. & Besdok, E., 2004. Impulsive noise suppression from images with the noise exclusive filter. *EURASIP Journal on Applied Signal Processing*, 16, pp.2434-40.

Cokal, C. & Solak, E., 2009. Cryptanalysis of a chaos-based image encryption algorithm. *Physics Letters A*, 373(15), pp.1357-60.

Corey, M., Farzam, F. & Chong, J.H., 2007. The effect of linearization of range in skin detection. In *Proceedings of IEEE International Conference on Information, Communications and Signal Processing*. Singapore, 2007. 10-13 December. pp. 1-5.

Costa, L. & Cesar, R., 2001. *Shape Analysis and Classification*. USA: CRC Press.

Couleur, 2008. [Online] Available at: HYPERLINK "http://www.couleur.org/index.php" http://www.couleur.org/index.php [Accessed 08 September 2009].

Cox, I., 2009. *Information hiding, watermarking and steganography*. Public Lecture. Londonderry: University of Ulster at Magee Intelligent Systems Research Centre.

CRYSTAL, 2004. *Cryptography and Encoding in the Context of Steganographic Algorithms*. [Online] Available at: HYPERLINK "http://www1.inf.tu-dresden.de/~aw4/crystal/slides.slide_1.html" http://www1.inf.tu-dresden.de/~aw4/crystal/slides.slide_1.html [Accessed 08 September 2009].

Deguillaume, F., Voloshynovskiy, S. & Pun, T., 2003. Secure hybrid robust watermarking resistant against tampering and copy attack. *Signal Processing*, 83(10), pp.2133-70.

Delahaye, J.P., 1996. Information noyée, information cache. *Pour la Science*, 229, pp.142-46. www.apprendre-en-ligne.net/crypto/stegano/229_142_146.pdf.

Denis, T.S., 2006. Cryptography for Developers. In *Syngress*., 2006.

Doherty, J., 2009. *Song form intelligence for streaming audio across wireless bursty networks*. PhD Thesis. Londonderry: University of Ulster School of Computing and Intelligent Systems.

Drew, M. & Bergner, S., 2007. *Spatio-chromatic decorrelation for color image compression*. Vancouver, Canada: http://fas.sfu.ca/pub/cs/TR/2007/CMPT2007-09.pdf Technical Report, Simon Fraser University.

El Safy, R.O., Zayed, H.H. & El Dessouki, A., 2009. An adaptive steganographic technique based on integer wavelet transform. In *International Conference on Networking and Media Convergence*., 2009. Egypt. pp. 111-117.

Encinas, L.H. & Dominguez, A.P., 2006. Comment on 'A technique for image encryption using digital signature'. *Optics Communications*, 268(2), pp.261-65.

Fard, A.M., Akbarzadeh-T, M. & Varasteh-A, F., 2006. A new genetic algorithm approach for secure JPEG steganography. In *Proceedings of IEEE International Conference on Engineering of Intelligent Systems*., 2006. 22-23 April. pp.1-6.

Farid, H., 2008. *Fundamentals of Image Processing*. [Online] Available at: HYPERLINK "http://www.cs.dartmouth.edu/farid/tutorials/" http://www.cs.dartmouth.edu/farid/tutorials/ [Accessed 21 July 2009].

Farid, H., 2009. A survey of image forgery detection. *IEEE Signal Processing Magazine*, 26(2), pp.16-25.

Floyd, R.W. & Steinberg, L., 1975. An Adaptive Algorithm for Spatial Gray Scale. In *Proceedings of International Symposium Digest of Technical*., 1975.

Ford, A. & Roberts, A., 1998. *Colour Space Conversions*. [Online] (b) Available at: HYPERLINK "http://www.poynton.com/PDFs/coloureq.pdf" http://www.poynton.com/PDFs/coloureq.pdf [Accessed 28 July 2009].

Forsyth, D. & Fleck, M., 1999. Automatic detection of human nudes. *International Journal of Computer Vision*, 32(1), pp.63-77.

Franz, E. & Schneidewind, A., 2004. Adaptive steganography based on dithering. In *Proceedings of the ACM Workshop on Multimedia and Security*. Magdeburg, Germany, 2004. September 20-21. pp.56-62.

Fridrich, J., 1997. *Secure image ciphering based on chaos*. Rome NY, USA: Final Report for AFRL.

Fridrich, J., 1999. Application of data hiding in digital images. In *Tutorial for the ISSPA'99*. Brisbane, Australia, 1999. August 22-25.

Fridrich, J. & Goljan, M., 1999b. Protection of Digital Images Using Self Embedding. In *Proceedings of Symposium on Content Security and Data Hiding in Digital Media*., 1999b.

Fridrich, J. & Goljan, M., 1999. Images with Self-Correcting Capabilities. In *Proceedings of International Conference on Image Processing*. Kobe, Japan, 1999. 24-28 October. pp.792-796.

Fridrich, J. & Goljan, M., 2002. Practical steganalysis of digital images-state of the art. In *Proceedings of SPIE Photonics West, Electronic Imaging'02, Security and Watermarking of Multimedia Contents*. San Jose, California, USA, 2002. January.

Fridrich, J., Goljan, M. & Du, R., 2001a. Detecting LSB steganography in color and gray-scale images. *IEEE Multimedia*, 8(4), pp.22-28.

Fridrich, J., Goljan, M. & Du, R., 2001b. Reliable detection of LSB steganography in grayscale and color images. In *Proceedings of ACM, Special Session on Multimedia Security and Watermarking*. Ottawa, Canada, 2001b. 5th Oct.

Fridrich, J., Goljan, M. & Hogeg, D., 2002. Steganalysis of JPEG images: Breaking the F5 algorithm. In *Proceedings of 5th International Workshop on Information Hiding, IH 2002*. Noordwijkerhout, The Netherlands, 2002. LNCS, Springer. 7-9 October. pp.310-323.

Fridrich, J., Goljan, M. & Soukal, D., 2005. Perturbed quantization steganography. *ACM Multimedia and Security Journal*, 11(2), pp.98-107.

Fridrich, J., Pevny, T. & Kodovsky, J., 2007. Statistically undetectable JPEG steganography: Dead ends, challenges, and opportunities. In *Proceedings of the ACM 9th Workshop on Multimedia & Security*. Dallas, Texas, USA, 2007. 20-21 September. pp.3-14.

Fridrich, J., Soukal, D. & Lukáš, J., 2003. Detection of Copy-Move Forgery in Digital Images. In *Proceedings of Digital Forensic Research Workshop*. Ohio, USA, 2003. 6-8 August.

Frith, D., 2007. Steganography approaches, options, and implications. *Network Security*, 8, pp.4-7.

Frontline Defenders, 2003. *Cryptology and Circumvention*. [Online] Available at: HYPERLINK "http://www.frontlinedefenders.org/manual/en/esecman/chapter3_4.html" http://www.frontlinedefenders.org/manual/en/esecman/chapter3_4.html [Accessed 21 July 2009].

Fu, M.S. & Au, O.C., 2002. Data hiding watermarking for halftone images. *IEEE Transactions on Image Processing*, 11(4), pp.477-84.

Gao, T. & Chen, Z., 2008. A new image encryption algorithm based on hyper-chaos. *Physics Letters A*, 372(4), pp.394-400.

Gomez, G., 2002. On selecting colour components for skin detection. In *Proceedings of International Conference on Pattern Recognition*. Quebec, Canada, 2002. 11-15 Aug. pp.961-964.

Gonzalez, R.C. & Woods, R.E., 2002. *Digital image processing*. Prentice Hall. ch. 8.

Gopinathan, U. et al., 2005. Strengths and weaknesses of optical encryption algorithms. In *Proc. 18th Annual Meeting of the IEEE Lasers and Electro-Optics Society*., 2005. 22-28 Oct.

Guo, J.M., 2008. Watermarking in dithered halftone images with embeddable cells selection and inverse halftoning. *Signal Processing*, 88(6), pp.1496-510.

Hashad, A.I., Madani, A.S. & Wahdan, A.E.M.A., 2005. A robust steganography technique using discrete cosine transform insertion. In *Proceedings of IEEE/ITI 3rd International Conference on Information and Communications Technology Enabling Technologies for the New Knowledge Society*. Cairo, Egypt, 2005. 5-6 Dec. pp. 255-264.

Hayati, P., Potdar, V. & Chang, E., 2007. A survey of steganographic and steganalytic tools for the digital forensic investigator. In *Proceedings of the Workshop of Information Hiding and Digital Watermarking in conjunction with IFIPTM*. New Brunswick, Canada, 2007. 30 July.

Hell, M., Johansson, T. & Brynielsson, L., 2009. An overview of distinguishing attacks on stream ciphers. *Cryptography and Communications*, 1(1), pp.71-94.

Hernandez-Castro, J.C., Blasco-Lopez, I. & Es, J.M., 2006. Steganography in games: A general methodology and its application to the game of Go. *Computers and Security*, 25, pp.64-71.

He, J., Zheng, J., Li, Z.B. & Qian, H.F., 2009. An improved colour image encryption based on chaotic map and OCML model. In *Proceedings of International Conference on Networks Security, Wireless Communications and Trusted Computing*. China, 2009. 25-26 Apr. pp. 365-369.

Hioki, H., 2002. A data embedding method using BPCS principle with new complexity measures. In *Proceedings of Pacific Rim Workshop on Digital Steganography*., 2002. July. pp. 30-47.

Ho, A.T.S. & Shu, F., 2003. A print-and-scan resilient digital watermark for card authentication. In *Proceedings of International Conference on Information, Communications and Signal Processing*. Singapore, 2003. 15-18 December. pp. 1149-1152.

Hosmer, C., 2006. Discovering hidden evidence. *Journal of Digital Forensic Practice*, (1), pp.47-56.

Hsieh, I.S., Fan, K.C. & Lin, C., 2002. A statistic approach to the detection of human faces in color nature scene. *Pattern Recognition*, 35(7), pp.1583-96.

Hsu, R.L., Abdel-Mottaleb, M. & Jain, A.K., 2002. Face detection in color images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(5), pp.696-702.

Huang, F. & Feng, Y., 2009. Security analysis of image encryption based on two dimensional chaotic maps and improved algorithm. *Journal of Frontiers of Electrical and Electronic Engineering in China*, 4(1), pp.5-9.

Huang, C.K. & Nien, H.H., 2009. Multi chaotic systems based pixel shuffle for image encryption. *Optics Communications*, 282(11), pp.2123-27.

Hu, J. & Han, F., 2009. A pixel-based scrambling scheme for digital medical images protection. *Journal of Network and Computer Applications*, 32(4), pp.788-94.

Hwang, R.J., Shih, K.T., Kao, C.H. & Chang, T.M., 2001. Lossy compression tolerant steganography. In *Proceedings of the 1st International Conference on The Human Society and the Internet – Internet Related Socio-Economic Issues*. Seoul, Korea, 2001. Lecture Notes In Computer Science. 4-6 July. pp. 427-435.

Jain, A.K. & Uludag, U., 2002. Hiding fingerprint minutiae in images. In *Proceedings of Workshop on Automatic Identification Advanced Technologies*. New York, USA, 2002. 7- 8 June. pp.97-102.

Jarvis, J.F., Judice, C.N. & Ninke, W.H., 1976. A Survey of Techniques for the Display of Continuous-tone Pictures on Bilevel Displays. *Computer Graphics and Image Processing*, 5(1), pp.13-40.

Jayaram, S., Schmugge, S., Shin, M.C. & Tsap, L.V., 2004. Effect of colorspace transformation, the illuminance component, and color modelling on skin detection. In *Proc of IEEE Computer Vision and Pattern Recognition (CVPR'04),*. Washington, USA, 2004. 27th June-2nd July.

Johnson, N.F. & Jajodia, S., 1998. Exploring steganography: Seeing the unseen. *IEEE Computer*, 31(2), pp.26-34.

Johnson, N.F. & Katzenbeisser, S.C., 2000. A survey of steganographic techniques. In Katzenbeisser, S. & Petitcolas, F.A.P. *Information hiding techniques for steganography and digital watermarking*. Norwood: Artech House, INC.

Joshi, M., Shakher, C. & Singh, K., 2008. Image encryption and decryption using fractional Fourier transform and radial Hilbert transform. *Optics and Lasers in Engineering*, 46(7), pp.522-26.

JPEG2000, 2007. *Our New Standard!* [Online] Available at: HYPERLINK "http://www.jpeg.org/jpeg2000/" http://www.jpeg.org/jpeg2000/ [Accessed 28 July 2009].

Judge, J.C., 2001. *Steganography: Past, Present, Future*. [Online] Available at: HYPERLINK "http://www.sans.org/reading_room/whitepapers/stenganography/steganography_past_present_future_552?show=552.php&cat=stenganography" http://www.sans.org/reading_room/whitepapers/stenganography/steganography_past_present_future_552?show=552.php&cat=stenganography [Accessed 21 July 2009].

Jung, K.H. & Yoo, K.Y., 2009. Data hiding method using image interpolation. *Computer Standards and Interfaces*, 31(2), pp.465-70.

Kahn, D., 1996. The Codebreakers: The comprehensive history of secret communication from ancient times to the Internet. *Scribner*.

Kanso, A. & Smaoui, N., 2009. Logistic chaotic maps for binary numbers generations. *Chaos, Solitons & Fractals*, 40(5), pp.2557-68.

Katzenbeisser, S.C., 2000. Principles of steganography. In Katzenbeisser, S. & Petitcolas, F.A.P. *Information hiding techniques for steganography and digital watermarking*. Norwood: Artech House, INC.

Kawaguchi, E. & Eason, R.O., 1998. Principle and applications of BPCS steganography. In *Proceedings of SPIE International Symposium on Voice, Video, and Data Communications*., 1998. 2-4 Nov.

Kermani, Z.Z. & Jamzad, M., 2005. A Robust Steganography Algorithm Based on Texture Similarity using Gabor Filter. In *Proceedings of IEEE 5th International Symposium on Signal Processing and Information Technology*., 2005. 18-21 Dec.

Khan, U.A., Cheema, M.I. & Sheikh, N.M., 2002. Adaptive video encoding based on skin tone region detection. In *Proceedings of IEEE Students Conference*. Pakistan, 2002. 16-17 August. pp. 129-134.

Kharrazi, M., Sencar, H.T. & Memon, N., 2006. Performance study of common image steganography and steganalysis techniques. *Journal of Electrical Imaging*, 15(4), pp.1-16.

Kodovsky, J. & Fridrich, J., 2008a. Influence of embedding strategies on security of steganographic methods in the JPEG domain. In *Proceedings of SPIE Electronic Imaging, Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*. San Jose, CA, USA, 2008a. January 28-30.

Kodovsky, J. & Fridrich, J., 2008b. On completeness of feature spaces in steganalysis. In *Proceedings of the 10th ACM workshop on Multimedia and security*. Oxford, UK, 2008b. 22-23 Sep. pp.123-132.

Kodovský, J. & Fridrich, J., 2009. Calibration revisited. In *ACM Multimedia and Security Workshop*. Princeton, NJ, USA, 2009.

Kong, J., Jia, H., Li, X. & Qi, Z., 2009. A novel content-based information hiding scheme. In *Proceedings of the International Conference on Computer Engineering and Technology*. Singapore, 2009. 22-24 Jan. pp. 436-440.

Kong, X., Wang, Z. & You, X., 2005. steganalysis of palette images: Attack optimal parity assignment algorithm. In *Proceedings of 5th IEEE International Conference on Information, Communications and Signal Processing*. Bangkok, Thailand, 2005. 06-09 Dec.pp.860-864.

Kostopoulos, I., Gilani, S.A.M. & Skodras, A.N., 2002. Colour image authentication based on a self-embedding technique. In *Proceedings of International Conference on Digital Signal Processing*. Greece, 2002. 1-3 July. pp.733-736.

Kovač, J., Peer, P. & Solina, F., 2003. Human skin colour clustering for face detection. In *Proceedings of International Conference on Computer as a Tool*. Slovenia, 2003. 22-24 September. pp.144-148.

Kruus, P., Scace, C., Heyman, M. & Mundy, M., 2003. A survey of steganographic techniques for image files. *Advanced Security Research Journal*, I, pp.41-51.

Kurak, C. & McHugh, J., 1992. A cautionary note on image downgrading. In *Proceedings of the IEEE 8th Annual Computer Security Applications Conference*. Texas, 1992. 30 Nov-4 Dec. pp.153-159.

Kutter, M. & Petitcolas, F., 1999. A fair benchmark for image watermarking systems. In *Proceedings of Electronic Imaging '99, Security and Watermarking of Multimedia Contents*. San Jose, California, USA, 1999. 25–27 January.

Kwok, H.S. & Tang, W.K.S., 2007. A fast image encryption system based on chaotic maps with finite precision representation. *Chaos, Solitons and Fractals*, 32(4), pp.1518-29.

L'Ecuyer, P., 2006. Uniform random number generation. In Henderson, S.G. & Nelson, B.L. *Handbook in Operations Research and Management Science*. Elsevier Science. p.57.

Lee, H.J. & Lee, C.C., 2005. *Human skin tone detection in YCbCr space*. [Online] USA: Patent, US 2005/0207643 A1.

Lee, S., Yoo, C.D. & Kalker, T., 2007. Reversible image watermarking based on integer-to-integer wavelet transform. *IEEE Transactions on Information Forensics and Security*, 2(3), pp.321-30.

Lian, S., Sun, J. & Wang, S.Z., 2005. A block cipher based on a suitable use of the chaotic standard map. *Chaos, Solitons and Fractals*, 26(1), pp.117-29.

Lian, N.X., Zagorodnov, V. & Tan, Y.P., 2006. Image denoising using optimal color space projection. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*. France, 2006. 14-19 May. pp.93-96.

Li, Z., Chen, X., Pan, X. & Zeng, X., 2009. Lossless data hiding scheme based on adjacent pixel difference. In *Proceedings of the International Conference on Computer Engineering and Technology*. Singapore, 2009. 20-22 Jan. pp.588-592.

Li, S., Li, C., Chen, G. & Lo, K.T., 2008b. Cryptanalysis of RCES/RSES image encryption scheme. *Journal of Systems and Software*, 81(7), pp.1130-43.

Li, Y., Li, C. & Wei, C., 2007. Protection of mammograms using blind steganography and watermarking. In *Proceedings of the IEEE International Symposium on Information Assurance and Security*., 2007.

Lin, C.Y. & Chang, S.F., 2001. SARI: Self-Authentication-and-Recovery Image Watermarking System. In *Proceedings of ACM International Conference on Multimedia*. Ontario, Canada, 2001. 30 Sept- 5 Oct.pp.628-629.

Lin, E.T. & Delp, E.J., 1999. A review of data hiding in digital images. In *Proceedings of the Image Processing, Image Quality, Image Capture Systems Conference, PICS'99*. Georgia, USA, 1999. the Society for Imaging Science and Technology. 25-28 April. pp.274-278.

Lin, C.C., Tai, W.L. & Chang, C.C., 2008. Multilevel reversible data hiding based on histogram modification of difference images. *Pattern Recognition*, 41(12), pp.3582-91.

Li, B., Shi, Y.Q. & Huang, J., 2008a. Steganalysis of YASS. In *Proceedings of 10th ACM Workshop on Multimedia and Security*. Oxford, United Kingdom, 2008a. 22-23 September. pp.139-148.

Liu, N., Guo, D. & Parr, G., 2006. A New Image Steganography for Internet Communications Based on Chaotic Sequences. In *International Conference on Intelligent Information Hiding and Multimedia Signal Processing*. Pasadena, CA, USA, 2006. 18-20 December. pp.121-124.

Liu, Y. & Wang, C.X., 2008. An improved algorithm of human skin detection in video image based on linear combination of 2-order Markov and Wiener predictor. In *Proceedings of International Symposium on Computer Science and Computational Technology*. Shanghai, China, 2008. 20-22 December.

Liu, P., Zhu, Z., Wang, H. & Yan, T., 2007. A novel image steganography using chaotic map and visual model. In *International Conference on Intelligent Systems and Knowledge Engineering (ISKE 2007)*. Chengdu, China, 2007. Atlantis Press.

Li, X. & Wang, J., 2007. A steganographic method based upon JPEG and particle swarm optimization algorithm. *Information Sciences*, 177(15), pp.3099-31091.

Li, X., Yuan, T., Yu, N. & Yuan, Y., 2003. Adaptive color quantization based on perceptive edge protection. *Pattern Recognition Letters*, 24(16), pp.3165-76.

Lou, D.C., Hu, M.C. & Liu, J.L., 2009. Multiple layer data hiding scheme for medical images. *Computer Standards and Interfaces*, 31(2), pp.329-35.

Lou, D.C. & Sung, C.H., 2004. A steganographic scheme for secure communications based on the chaos and Euler theorem. *IEEE Transactions on Multimedia*, 6(3), pp.501-09.

Lukáš, J., Fridrich, J. & Goljan, M., 2006. Detecting digital image forgeries using sensor pattern noise. In *Proceedings of SPIE, Security, Steganography, and Watermarking of Multimedia Contents VIII.*, 2006.

Luo, H., Chu, S.C. & Lu, Z.M., 2008. Self Embedding Watermarking Using Halftoning Technique. *Circuits Systems and Signal Processing*, 27, pp.155-70.

Lyu, S. & Farid, H., 2006. Steganalysis using higher-order image statistics. *IEEE Transactions on Information Forensics and Security*, 1(1), pp.111-19.

Maity, S.P., Kundu, M.K. & Nandi, P.K., 2004. Genetic algorithm for optimal imperceptibility in image communication through noisy Channel. In *International Conference on Neural Information Processing (ICONIP '2004)*. India, 2004. LNCS. 29 October. pp.700-705.

Maniccam, S.S. & Bourbakis, N.G., 2004. Image and video encryption using SCAN patterns. *Pattern Recognition*, 37(4), pp.725-37.

Manikopoulos, C. et al., 2002. Detection of block DCT-based steganography in gray-scale images. In *Proceedings of the IEEE Workshop on Multimedia Signal Processing*. US Virgin Islands, 2002. 9-11 Dec. pp.355-358.

Mao, Y. & Wu, M., 2006. A joint signal processing and cryptographic approach to multimedia encryption. *IEEE Transactions on Image Processing*, 15(7), pp.2061-75.

Martinian, E., Yekhanin, S. & Yedidia, J.S., 2005. Secure biometrics via syndromes. In *Proceedings of the Allerton Conference on Communication, Control, and Computing*. Monticello, USA, 2005. Mitsubishi Electric Research Laboratories. October.

Martinkauppi, J.B., Soriano, M.N. & Laaksonen, M.H., 2001. Behavior of skin color under varying illumination seen by different cameras at different color spaces. In *Proc. of SPIE, Machine Vision Applications in Industrial Inspection IX*. USA, 2001.

Martin, A., Sapiro, G. & Seroussi, G., 2005. Is image steganography natural? *IEEE Transactions on Image Processing*, 14(12), pp.2040-50.

Marvel, L.M. & Retter, C.T., 1998. A methodology for data hiding using images. In *Proceedings of IEEE Military Communications Conference, MILCOM'98*. Boston, MA, USA, 1998. 18-21 Oct. pp.1044-1047.

Mazloom, S. & Eftekhari-Moghadam, A.M., 2009. Color image encryption based on Coupled Nonlinear Chaotic Map. *Chaos, Solitons and Fractals*, 42(3), pp.1745-54.

McGill, L., 2005. *Steganography: the right way*. SANS Institute InfoSec Reading Room.

McKeon, R.T., 2007. Strange Fourier steganography in movies. In *Proceedings of the IEEE International Conference on Electro/Information Technology (EIT)*. Chicago IL, USA, 2007. 17-20 May. pp.178-182.

Menezes, A.J., van Oorschot, P.C. & Vanstone, S.A., 1996. *Handbook of Applied Cryptography*. CRC Press. 175-177, ch. 5.

Mese, M. & Vaidyanathan, P.P., 2001. Look-up table (LUT) method for inverse halftoning. *IEEE Transactions on Image Processing*, 10(10), pp.1566-78.

Miaou, S., Hsu, C., Tsai, Y. & Chao, H., 2000. A secure data hiding technique with heterogeneous data-combining capability for electronic patient records. In *Proceedings of the IEEE 22nd Annual EMBS International Conference*. Chicago, USA, 2000. 23-28 July. pp.280-283.

Miller, M.L., Doërr, G.J. & Cox, I.J., 2004. Applying Informed Coding and Embedding to Design a Robust High-Capacity Watermark. *IEEE Transactions on Image Processing*, 13(6), pp.792-807.

Mitra, A., Subba Rao, Y.V. & Prasanna, S.R.M., 2006. A new image encryption approach using combinational permutation techniques. *International Journal of Computer Science*, 1(2), pp.127-31.

Moulin, P. & Koetter, R., 2005. Data-hiding codes. *Proceedings of the IEEE*, 93(12), pp.2083-126.

Murtagh, F., 2007. The Haar wavelet transform of a dendrogram. *Journal of Classification*, 24(3-32), pp.1-38.

Nakamura, H. & Zhao, Q., 2008. Information hiding based on image morphing. In *Proceedings of 22nd International Conference on Advanced Information Networking and Applications Workshops, AINAW*. CA, USA, 2008. 25-28 March. pp.1585-1590.

National-Archive, 2008. *Forged documents*. [Online] Available at: HYPERLINK "http://www.nationalarchives.gov.uk/news/stories/195.htm?homepage=news" http://www.nationalarchives.gov.uk/news/stories/195.htm?homepage=news [Accessed 21 July 2009].

Neelamani, R., Nowak, R. & Baraniuk, R., 2000. Model-Based Inverse Halftoning with Wavelet-Vaguelette Deconvolution. In *Proceedings of International Conference on Image Processing*. BC, Canada, 2000. 10-13 September. pp.973-976.

Neelamani, R., Nowak, R. & Baraniuk, R., 2009. WInHD: Wavelet-based Inverse Halftoning via Deconvolution. *Rejecta Mathematica*, 1(1), pp.84-103.

Nikolaidis, A. & Pitas, I., 2000. Robust watermarking of facial images based on salient geometric pattern matching. *IEEE Transactions on Multimedia*, 2(3), pp.172-84.

Nikolaidis, A. & Pitas, I., 2001. Region-based image watermarking. *IEEE Transactions on Image Processing*, 10(11), pp.1726-40.

Nirinjan, U.C. & Anand, D., 1998. Watermarking medical images with patient information. In *Proceedings of the 20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. Hong Kong, China, 1998. 29 Oct-1 Nov. pp.703-706.

Online Software, n.d. [Online] Available at:
[Camouflage]: http://camouflage.unfiction.com/
[Data Stash]: http://www.skyjuicesoftware.com/software/ds_info.html
[F5]: http://wwwrn.inf.tu-dresden.de/~westfeld/f5.html
[Hide and Seek]:
ftp://ftp.funet.fi/pub/crypt/mirrors/idea.sec.dsi.unimi.it/cypherpunks/steganography/hdsk41b.zip
[Hide in Picture]: http://sourceforge.net/projects/hide-in-picture/
[JpegX]: http://www.freewarefiles.com/Jpegx_program_19392.html
[Other Tools]: http://www.jjtc.com/Security/stegtools.htm
[OutGuess]: http://www.outguess.org/
[Revelation]: http://revelation.atspace.biz/
[Stella] : http://wwwicg.informatik.uni-rostock.de/~sanction/stella/
[S-Tools]: ftp://ftp.funet.fi/pub/crypt/mirrors/idea.sec.dsi.unimi.it/code/s-tools4.zip
O'Ruanaidh, J.J.K. & Pun, T., 1997. Rotation, scale and translation invariant digital image watermarking. In *Proceedings of IEEE International Conference on Image Processing (ICIP 97)*. Santa Barbara, CA, USA, 1997. 26-29 October. pp.536-539.

Patidar, V., Pareek, N.K. & Sud, K.K., 2009. A new substitution-diffusion based image cipher using chaotic standard and logistic maps. *Communications in Nonlinear Science and Numerical Simulation*, 14(7), pp.3056-75.

Paulson, L.D., 2006. New system fights steganography. *News Briefs*, 39(8), pp.25-27.

Peng, Z. & Liu, W., 2008. Color image authentication based on spatiotemporal chaos and SVD. *Chaos Solitons and Fractals*, 36(4), pp.946-52.

Petitcolas, F.A.P., 2000. Introduction to information hiding. In Katzenbeisser, S. & Petitcolas, F.A.P. *Information hiding techniques for steganography and digital watermarking*. Norwood: Artech House, INC.

Pevny, T. & Fridrich, J., 2007. Merging Markov and DCT features for multi-class JPEG steganalysis. In *Proceedings of SPIE Electronic Imaging*. Photonics West, USA, 2007. January.

Phung, S.L., Bouzerdoum, A. & Chai, D., 2005. Skin segmentation using color pixel classification: analysis and comparison. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(1), pp.148-54.

Pisarchik, A.N., Flores-Carmona, N.J. & Carpio-Valadez, M., 2006. Encryption and decryption of images with chaotic map lattices. *Chaos*, 16(033118).

Popescu, A.C., 2005. *Statistical tools for digital image forensics*. USA: http://www.cs.dartmouth.edu/~farid/publications/apthesis05.html Ph.D. Dissertation, Dartmouth College. Date accessed: 16-05-07.

Porle, R.R., Chekima, A., Wong, F. & Sainarayanan, G., 2007. Wavelet-based skin segmentation for detecting occluded arms in human body pose modelling system. In *Proceedings of International Conference on Intelligent and Advanced Systems*. Malaysia, 2007. 25-28 November. pp.764-769.

Potdar, V.M., Han, S. & Chang, E., 2005a. A survey of digital image watermarking techniques. In *Proceedings of the IEEE 3rd International Conference on Industrial Informatics (INDIN)*. Perth, Australia, 2005a. 10-12 August. pp.709-716.

Potdar, V.M., Han, S. & Chang, E., 2005b. Fingerprinted secret sharing steganography for robustness against image cropping attacks. In *Proceedings of IEEE 3rd International Conference on Industrial Informatics (INDIN)*. Perth, Australia, 2005b. 10-12 August.pp.717-724.

Provos, N., 2001. *Defending against statistical steganalysis*. USA: Technical report, University of Michigan.

Provos, N. & Honeyman, P., 2001. *Detecting steganographic content on the Internet*. Technical report, University of Michigan. August 31.

Provos, N. & Honeyman, P., 2003. Hide and seek: An introduction to steganography. *IEEE Security and Privacy*, 01(3), pp.32-44.

Puri, A. & Eleftheriadis, A., 1998. MPEG-4: An object-based multimedia coding standard supporting mobile applications. *Mobile Networks and Applications*, 3(1), pp.5-32.

Raja, K.B., Chowdary, C.R., Venugopal, K.R. & Patnaik, L.M., 2005. A secure image steganography using LSB, DCT and compression techniques on raw images. In *Proceedings of*

IEEE 3rd International Conference on Intelligent Sensing and Information Processing, ICISIP'05. Bangalore, India, 2005. 14-17 Dec. pp.170-176.

Raja, K.B. et al., 2008. Robust image adaptive steganography using integer wavelets. In *Proceedings of the 3rd International Conference on Communication Systems Software and Middleware and Workshops, COMSWARE'08*. Bangalore, India, 2008. 5-10 Jan. pp.614-621.

Raja, K.B., Vikas, Venugopal, K.R. & Patnaik, L.M., 2006. High capacity lossless secure image steganography using wavelets. In *Proceedings IEEE International Conference on Advanced Computing and Communications (ADCOM 2006)*. India, 2006. 20-23 Dec. pp.230-235.

Ramani, K., Prasad, E.V. & Varadarajan, S., 2007. Steganography using BPCS to the integer wavelet transformed image. *International Journal of Computer Science and Network Security*, 7(7), pp.293-302.

Refregier, P. & Javidi, B., 1995. Optical image encryption based on input plane and Fourier plane random encoding. *Optics Letters*, 20(7), pp.767-69.

Rhouma, R. & Belghith, S., 2008. Cryptanalysis of a new image encryption algorithm based on hyper-chaos. *Physics Letters A*, 372(38), pp.5973-78.

Rodriguez, A. & Rowe, L., 1995. Multimedia systems and applications. *IEEE Computer*, 28(5), pp.20-22.

Rosenbaum, R. & Schumann, H., 2000. A steganographic framework for reference colour based encoding and cover image selection. In *Proceedings of 3rd International Workshop on Information Security (ISW 2000)*. Wollongong, Australia, 2000. LNCS, Springer. 20-21 December. pp.415-434.

Rukhin, A. et al., 2008. special publication 800-22 *A statistical test suite for random and pseudorandom number generators for cryptographic applications*. National Institute of Standards and Technology (NIST).

Saenz, M., Oktem, R., Egiazarian, K. & Delp, E., 2000. Color image wavelet compression using vector morphology. In *Proceedings of the European Signal Processing Conference*. Tampere, Finland, 2000. 5-8 September. pp.5-8.

Sallee, P., 2003. Model-based steganography. In *Proceedings of the 2nd International Workshop on Digital Watermarking*. Seoul, Korea, 2003. LNCS. 20-22 October. pp.254-260.

Sallee, P., 2005. Model-based methods for steganography and steganalysis. *International Journal of Image and graphics*, 5(1), pp.167-90.

Scottish Radiological Society, 2002. *Lung Case Index*. [Online] Available at: HYPERLINK "http://www.radiology.co.uk" http://www.radiology.co.uk [Accessed 28 July 2009].

SHA, 2001. *US Secure Hash Algorithm 1*. [Online] Available at: HYPERLINK "http://www.faqs.org/rfcs/rfc3174/" http://www.faqs.org/rfcs/rfc3174/ [Accessed 21 July 2009].

Shaik, Z. & Asari, V., 2007. A robust method for multiple face tracking using Kalman filter. In *Proceedings of IEEE Applied Imagery Pattern Recognition Workshop*. Washington, USA, 2007. 10-12 October. pp.125-130.

Shannon, C.E., 1949. Communication theory of secrecy systems. *Bell System Technical Journal*, 28(4), pp.656-715.

Shao, Y., Zhang, L., Wu, G. & Lin, X., 2001. Reconstruction of Missing Blocks In Image Transmission by Using Self-Embedding. In *Proceedings of International Symposium on Intelligent Multimedia, Video and Speech Processing*., 2001.

Shefali, S., Deshpande, S.M. & Tamhankar, S.G., 2008. Attack Detection through Image Adaptive Self Embedding Watermarking. *International Journal of Signal Processing*, 4(4), pp.260-66.

Shih, F., 2008. *Digital watermarking and steganography, fundamentals and techniques*. USA: CRC Press.

Shin, M.C., Chang, K.I. & Tsap, L.V., 2002. Does colorspace transformation make any difference on skin detection? In *Proceedings of IEEE Workshop on Applications of Computer Vision*. Florida, USA, 2002. 3-4 December. pp.275-279.

Shin, C.M. & Kim, S.J., 2006. Phase-only encryption and single path decryption system using phase-encoded exclusive-OR rules in Fourier domain. *Optical Review*, 13(2), pp.49-52.

Shirali-Shahreza, M.H. & Shirali-Shahreza, M., 2006. A new approach to Persian/Arabic text steganography. In *Proceedings of 5th IEEE/ACIS International Conference on Computer and Information Science (ICIS-COMSAR 2006)*. Honolulu, Hawaii, 2006. 10-12 July. pp.310-315.

Shujun, L., Guanrong, C. & Xuan, Z., 2004. Chaos-based encryption for digital images and videos. In Furht, B. & Kirovski, D. *Multimedia Security Handbook*. Boca Raton, FL, USA: CRC Press. pp.133-67.

Signal Processing Group, 2008. *Error Resilience & Transport*. [Online] Available at: HYPERLINK "http://www.bristol.ac.uk/eeng/research/spr/error.html" http://www.bristol.ac.uk/eeng/research/spr/error.html [Accessed 08 July 2009].

Silva, E.S. & Agaian, S., 2004. The Best Transform in the Replacement Coefficients and the Size of the Payload Relationship Sense. In *Proceedings of Society for Imaging Science & Technology*., 2004.

Simmons, G.J., 1984. The prisoners' problem and the subliminal channel. In *Proceedings of International conference on Advances in Cryptology,CRYPTO83*. New York, USA, 1984. August 22-24. pp.51-67.

Singh, M., Kumar, A. & Singh, K., 2008. Encryption and decryption using a sandwich phase diffuser made by using two speckle patterns and placed in the Fourier plane: Simulation results. *Optik - International Journal for Light and Electron Optics, article in press*.

Sinha, A. & Singh, K., 2003. A technique for image encryption using digital signature. *Optics Communications*, 218(4-6), pp.229-34.

Sinha, A. & Singh, K., 2006. Reply to comment on ‘A technique for image encryption using digital signature’. *Optics Communications*, 268(2), pp.266-68.

Siwei, L., 2005. *Natural image statistics for digital image forensics*. Thesis of Doctor of Philosophy. Hanover, New Hampshire, USA: Dartmouth College.

Socek, D. et al., 2007. New approaches to encryption and steganography for digital videos. *Multimedia Systems*, 13(3), pp.191-204.

Socek, D., Li, S., Magliveras, S.S. & Furht, B., 2005. Enhanced 1-D chaotic key-based algorithm for image encryption. In *Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks*., 2005. 05-09 Sept. pp.406-407.

Solak, E. & Çokal, C., 2008. Comment on ‘Encryption and decryption of images with chaotic map lattices. *Chaos*, 18(3), pp.038101-038101-3.

Solanki, K., Madhow, U. & Manjunath, B.S., 2006. 'Print and scan' resilient data hiding in images. *IEEE Transactions on Information Forensics and Security*, 1(4), pp.464-78.

Solanki, K., Sarkar, A. & Manjunath, B.S., 2007. YASS: Yet another steganographic scheme that resists blind steganalysis. In *Proceedings of the 9th International Workshop on Information Hiding*. Saint Malo, France, 2007. LNCS. 11-13 June.pp.16-31.

Spaulding, J., Noda, H., Shirazi, M.N. & Kawaguchi, E., 2002. BPCS steganography using EZW lossy compressed images. *Pattern Recognition Letters*, 23(13), pp.1579-87.

Srinivasan, Y., 2003. *High capacity data hiding system using BPCS steganography*. USA: http://etd.lib.ttu.edu/theses/available/etd-06272008-31295018922590/unrestricted/31295018922590.pdf Master Dissertation, Texas Tech. University. December.

Srinivasan, Y. et al., 2004. Secure transmission of medical records using high capacity steganography. In *Proceedings of the 17th IEEE Symposium on Computer-Based Medical Systems, CBMS’04*., 2004.

Starck, J.L., Fadili, J. & Murtagh, F., 2007. The undecimated wavelet decomposition and its reconstruction. *IEEE Transactions on Image Processing*, 16(2), pp.297-309.

Starck, J.L., Murtagh, F. & Bijaoui, A., 1998. *Image processing and data analysis: the multiscale approach*. 1st ed. Cambridge University Press.

Tarrés, F. & Rama, A., n.d. *GTAV Face Database*. [Online] Available at: HYPERLINK "http://gps-tsc.upc.es/GTAV/ResearchAreas/UPCFaceDatabase/GTAVFaceDatabase.htm" http://gps-tsc.upc.es/GTAV/ResearchAreas/UPCFaceDatabase/GTAVFaceDatabase.htm [Accessed 08 September 2009].

Thomas, T.L., 2003. Al Qaeda and the Internet: The danger of "cyberplanning". *Parameters*, Spring. www.carlisle.army.mil/usawc/Parameters/03spring/thomas.pdf.

Tian, J., 2003. Reversible data embedding using a difference expansion. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(8), pp.890-96.

Tong, X., Cui, M. & Wang, Z., 2009. A new feedback image encryption scheme based on perturbation with dynamical compound chaotic sequence cipher generator. *Optics Communications*, 282(14), pp.2722-28.

Tsai, P., Hu, Y.C. & Yeh, H.L., 2009. Reversible image hiding scheme using predictive coding and histogram shifting. *Signal Processing*, 89(6), pp.1129-43.

Tzschoppe, R., Baum, R., Huber, J. & Kaup, A., 2003. Steganographic system based on higher-order statistics. In *Proceedings of SPIE, Security and Watermarking of Multimedia Contents V*. Santa Clara, California, USA, 2003.

Ullerich, C. & Westfeld, A., 2007. Weaknesses of MB2. In *Proceedings of the 6th International Workshop on Digital Watermarking*. Guangzhou, China, 2007. 3-5 December.pp.127-142.

Usman, K. et al., 2007. Medical image encryption based on pixel arrangement and random permutation for transmission security. In *Proceedings of IEEE 9th International Conference on e-Health Networking, Application and Services*. Taipei, Taiwan, 2007. 19-22 June. pp.244-247.

Van Der Weken, D., Nachtegael, M. & Kerre, E., 2004. Using similarity measures and homogeneity for the comparison of images. *Image and Vision Computing*, 22(9), pp.695-702.

Verma, B., Jain, S. & Agarwal, D.P., 2005. Watermarking image databases: A review. In *Proceedings of the International Conference on Cognition and Recognition*. Mandya, Karnataka, India, 2005. 22-23 Dec. pp.171-179.

Vezhnevets, V., Sazonov, V. & Andreeva, A., 2003. A survey on pixel-based skin color detection techniques. In *Proc. Graphicon*. Moscow, 2003. September.

Wang, K.W., 2009. Image encryption using chaotic maps. In Kocarev, L., Galias, Z. & Lian, S. *Intelligent computing based on chaos*. Springer. p.345.

Wang, Y. et al., 2007. image encryption method based on chaotic map. In *Proceedings of IEEE 2nd Conference on Industrial Electronics and Applications (ICIEA)*. harbin, China, 2007. 23-25 May. pp.2558-2560.

Wang, Y., Xiaofeng, L., Di, X. & Kwok-Wo, W., 2008. One-way hash function construction based on 2D coupled map lattices. *Information Sciences*, 178(5), pp.1391-406.

Wang, X. & Zhang, J., 2008. An image scrambling encryption using chaos-controlled Poker shuffle operation., 2008.

Wayner, P., 2002. *Disappearing cryptography*. 2nd ed. Morgan Kaufmann Publishers.

Wen, J., Severa, M. & Zeng, W., 2002. A format-compliant configurable encryption framework for access control of video. *IEEE Trans. Circuits Syst. Video Technol*, 12(6), p.545–557.

Westfeld, A., 2001. F5-A steganographic algorithm: High capacity despite better steganalysis. In *Proceedings of 4th International Workshop on Information Hiding*. Pittsburgh, USA, 2001. LNCS. 25-27 April. pp.289-302.

Westfeld, A. & Pfitzmann, A., 1999. Attacks on steganographic systems breaking the steganography utilities EzStego, Jsteg, Steganos and S-Tools and some lessons learned. In *Proceedings of 3rd International Workshop on Information Hiding*. Dresden, Germany, 1999. LNCS. 29 September- 1 October. pp.61-76.

WolframMathWorld, 1999. *Gray Code*. [Online] Available at: HYPERLINK "http://mathworld.wolfram.com/GrayCode.html" http://mathworld.wolfram.com/GrayCode.html [Accessed 08 September 2009].

Wong, K.W., Kwok, B.S. & Law, W.S., 2008a. A fast image encryption scheme based on chaotic standard map. *Physics Letters A*, 372(15), pp.2645-52.

Wong, K.W., Kwok, B.S.H. & Yuen, C.H., 2008b. An efficient diffusion approach for chaos-based image encryption. *Chaos, Solitons and Fractals*, 45(5), pp.2652-63.

Wong, K.W., Lam, K.M. & Siu, W.C., 2003. A robust scheme for live detection of human faces in color images. *Signal Processing: Image Communication*, 18(2), pp.103-14.

Wu, Y.T. & Shih, F.Y., 2006. Genetic algorithm based methodology for breaking the steganalytic systems. *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, 36(1), pp.24-31.

Wu, D.C. & Tsai, W.H., 2003. A steganographic method for images by pixel-value differencing. *Pattern Recognition Letters*, 24(9-10), pp.1613-26.

Yekkala, A.K., Udupa, N., Bussa, N. & Madhavan, C.E.V., 2007. Lightweight encryption for images. In *Proceedings of International Conference on Consumer Electronics (ICCE 2007)*. Las Vegas, USA, 2007. 12-14 January. pp.1-2.

Yen, J.C. & Guo, J.I., 2000. A new chaotic key-based design for image encryption and decryption. In *Proc. of IEEE International Symposium on Circuits and Systems*. Geneva, Switzerland, 2000. May 28-31.

Yu, Y.H., Chang, C.C. & Lin, I.C., 2007. A new steganographic method for color and grayscale image hiding. *Computer Vision and Image Understanding*, 107(3), pp.183-94.

Yun, J.U., Lee, H.J., Paul, A.K. & Baek, J.H., 2007. Robust face detection for video summary using illumination-compensation and morphological processing. In *Proceedings of IEEE International Conference on Natural Computation*. China, 2007. 24-27 August. pp.710-714.

Yu, L., Zhao, Y., Ni, R. & Zhu, Z., 2009. PM1 steganography in JPEG images using genetic algorithm. *Soft Computing*, 13(4), pp.393-400.

Zeghid, M. et al., 2006. A modified AES based algorithm for image encryption. *International Journal of Computer Science and Engineering*, 1(1), pp.70-75.

Zeki, A.M. & Manaf, A.A., 2009. A novel digital watermarking technique based on ISB (Intermediate Significant Bit). *World Academy of Science, Engineering and Technology*, 38, pp.1080-87.

Zhang, Z. & Shi, Y., 2008. Face detection method based on a new nonlinear transformation of color spaces. In *Proceedings of International Conference on Fuzzy Systems and Knowledge Discovery*. China, 2008. 18-20 October. pp.34-38.

Zhang, X. & Wang, S., 2004. Vulnerability of pixel-value differencing steganography to histogram analysis and modification for enhanced security. *Pattern Recognition Letters*, 25(3), pp.331-39.

Zhao, X., Boussaid, F. & Bermak, A., 2007. Characterization of a 0.18 μ m CMOS color processing scheme for skin detection. *IEEE Sensors Journal*, 7(11), pp.1471-74.

Zhao, Y.J., Dai, S.L. & Xi, X., 2008. A Mumford-Shah level-set approach for skin segmentation using a new color space system. In *Proceedings of International Conference on Simulation and Scientific Computing*. China, 2008. 10-12 October. pp.307-310.

Zhao, Z., Yu, N. & Li, X., 2003. A novel video watermarking scheme in compression domain based on fast motion estimation. In *Proceedings of IEEE International Conference on Communication Technology*. Beijng, China, 2003. 9-11 April. pp.1878-1882.

Zheng, L. & Cox, I., 2007. JPEG based conditional entropy coding for correlated steganography. In *Proceedings of IEEE International Conference on Multimedia and Expo*. Beijing, China, 2007. 2-5 July. pp.1251-1254.

Zou, D., Tian, J., Bloom, J. & Zhai, J., 2006. Data Hiding in Film Grain. In *Proceedings of 5th International Workshop on Digital Watermarking*. Jeju Island, Korea, 2006. 8-10 November. pp.197-211.

Zou, J., Xiong, C., Qi, D. & Ward, R.K., 2005. The application of chaotic maps in image encryption. In *Proceedings of IEEE 3rd Northeast Workshop on Circuits and Systems NEWCAS*. Québec, Canada, 2005. 19-22 June. pp.331-334.