

AUTOMATIC DETECTION OF STEGANOGRAPHIC IMAGES TO FACILITATE DECODING



MCS

By

Muhammad Sadiq Khan

A thesis submitted to the faculty of Information Security Department, Military College
of Signals, National University of Sciences and Technology, Rawalpindi in partial
fulfillment of the requirements for the degree of MS in Information Security

Oct 2017

CERTIFICATE

This is to certify that **Muhammad Sadiq Khan** Student of **MSIS-14** Course Reg.No: **00000118874** has completed her MS Thesis title “**Automatic Detection of Steganographic Images to Facilitate Decoding**” under my supervision. I have reviewed her final thesis copy and am satisfied with her work.

Thesis Supervisor
(Col. Abdul Ghafoor, PHD)

Dated: _____

DEDICATION

“In the name of Allah, the most Beneficent, the most Merciful”

I dedicate this thesis to my teachers, parents, brothers and my wife and kids.

ACKNOWLEDGMENTS

First and foremost, I would like to thank Allah Almighty for providing me courage and motivation during the thesis to handle all challenges in a pleasing manner.

I thank my supervisor Col Dr. Abdul Ghafoor (HoD) for his kind support, availability and useful ideas that helped me to refine the research work.

I offer the deepest gratitude to my committee member supervisor Col Dr. Monis Akhlaq who has put his prodigious efforts throughout the thesis phase with his knowledge, expertise, and valuable suggestions. He has provided full support, mentorship, and continuous assistance despite his utmost busy commitments.

I also would like to thank my committee member Lec Narmeen Shafqat for her kind support and availability throughout my research.

I am very blessed to have my parents and my wife whose great company adds a significant contribution to my life and education. I would especially acknowledge them for their prayers and support without which I could not be able to achieve all this.

I would also appreciate my friends and colleagues Waqar Azam, Muhammad Awais, Umair Aslam, Alamzeb Amir and Aamir Raza for sharing their knowledge, insights and providing feedback. Our combined interactive sessions and discussions come up with the result that helped me to resolve the critical issues during the research phase.

And finally, thanks to my family, and numerous friends who endured this process with me, always offering support and care.

ABSTRACT

Undeniably we are living in digital age. Digital gadgets are creating millions of trillion digital documents each day. Techniques and methodologies of hiding classified digital data have changed now. Terrorists and bad guys changed their techniques to hide their confidential data. Encryption converts plain text into cipher text, which is directly unreadable, but encrypted data gives an idea to investigator that this data have some important information. Steganography is used for data hiding and communication of classified information un-noticeably. In image steganography, user hides the classified message behind an innocent or benign image. Sometimes user also modifies or removes file extension to further dodge the investigator. Investigation techniques are rated with its capacity (size) to hide secret message and imperceptibility. Several sophisticated techniques of image steganography are developed.

Steganalysis is the method to recover secret message embedded in benign image. Several software tools are available that develop their own stenography and Steganalysis technique. Poor encoding techniques can be reversed with some virtual or statistical attacks.

The purpose of this thesis is to develop a hybrid and sophisticated Steganalysis tool that decodes the embedded message encoded by common and best-practiced technique. This hybrid algorithm has the advantage to scan a complete drive or folder to recover the modified or removed file extension and decode if there is any steno-graphic image found. The developed tool contains the decoding algorithm of most common stenographic technique.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	i
ABSTRACT	ii
LIST OF FIGURE	vi
LIST OF TABLE	vii
KEY TO ACRONYMS	viii
1 INTRODUCTION	1
1.1 Introduction.....	1
1.2 Overview.....	1
1.3 History of data hiding techniques	2
1.3.1 Data Hiding - Steganography.....	2
1.3.2 Data Hiding – Cryptography.....	2
1.3.3 Data Hiding - Watermarking	3
1.4 Application of Image Steganography	4
1.5 Problem Statement.....	5
1.6 Advantageous stockholders of research.....	7
1.7 Proposed solution or Contribution	7
1.8 Conclusion	8
2 LITERATURE REVIEW	9
2.1 Introduction.....	9
2.2 Literature Review.....	9
2.3 Computer File Extension – Microsoft Windows Operating System	10
2.3.1 Tools and resourcing to read file signature.....	11
2.4 Data Hiding Methods.....	12
2.4.1 Steganography in Audio	13
2.4.2 Steganography in Video.....	13
2.4.3 Steganography in Images.....	13
2.5 Image Steganography – Fidelity Criteria.....	17
2.5.1 Robustness	17
2.5.2 Imperceptibility – Perceptual Transparency	18

2.5.3	Capacity	18
2.5.4	Security	18
2.6	Computational Cost	18
2.7	Image Steganography Techniques	18
2.8	Spacial Domain - Image Steganography.....	19
2.9	Transform Domain Image Steganography.....	20
2.10	Steganalysis – Steganography Detection Techniques.....	22
2.11	Cataloguing of Steganalysis techniques.....	23
2.11.1	Signature-based Steganography detection	23
2.11.2	Statistical based Steganography detection	24
2.12	Conclusion	25
3	DESIGN AND METHODOLOGY	26
3.1	Introduction.....	26
3.2	Design Goal	26
3.3	Existing similar programs and their limitations.....	27
3.3.1	VSL (Virtual Steganography Library)	27
3.3.2	StegDetect	27
3.4	F-5 Steganography Algorithm	28
3.5	Hybrid Steganalysis program.....	28
3.6	Design Features.....	28
3.7	Design Framework.....	29
3.8	Design Architecture / Platform	30
3.8.1	MATLAB.....	30
3.8.2	Microsoft Visual Studio.....	31
3.9	Design Parameters	31
3.9.1	Steganalysis - Extracting Secret Message.....	31
3.9.2	Scan for Modified or Removed file extension.....	32
3.9.3	Embedding Secret Message - (Steganography)	33
3.10	List of common file types.	34
3.11	Methodology	35
3.12	Conclusion	36

4	IMPLEMENTATION AND TESTING.....	37
4.1	Introduction.....	37
4.2	Forensic assessment / evaluation of Steganalysis tool.....	37
4.3	Scan result for modified or removed file extensions	37
4.4	LSB Steganography detection.....	38
4.5	Detect Steganography with SSDetect	40
4.6	SSDetect Comparison with other tools.....	42
4.7	Detect altered file extension and Stegno images at same time	44
4.8	Image encoding – Steganography.....	44
4.9	Working of hybrid stego detect model.....	45
4.10	Conclusion	48
5	CONCLUSION AND FUTURE DIRECTION	49
5.1	Introduction.....	49
5.2	Research flow.....	49
5.3	Research limitation	50
5.4	Future Direction	50
5.5	Conclusion	53
	APPENDICES	54
	Appendix I: DIRECTORY OF STEGANOGRAPHY SOFTWARE	54
	Appendix II: TECHNICAL COMPARISON OF IMAGE FORMATS	56
	Appendix III: SIGNATURE / MAGIC BYTES OF COMMON FILES	58
6	BIBLIOGRAPHY	59

LIST OF FIGURES

Figure 1-1 Data Hiding Techniques	4
Figure 1-2 Example of Image StegnoSploit	5
Figure 2-1 Magic Word	11
Figure 2-2 General Image Steganography Diagram	13
Figure 2-3 Text File Injection	15
Figure 2-4 Text File Extraction	16
Figure 2-5 Exif Data Hiding	17
Figure 2-6 Image Steganography Techniques	19
Figure 2-7 Image Steganography in Spatial Domain	19
Figure 2-8 Transform Domain Image Steganography	21
Figure 2-9 Steganalysis Techniques	22
Figure 3-1 SSDetect Framework	30
Figure 3-2 Extracting Secret Message	31
Figure 3-3 Agile Software Development Model	35
Figure 4-1 Contingency Table	39
Figure 4-2 Performance Analysis of the LSB detection	39
Figure 4-3 Scan Results for Universal Steganography Detection	41
Figure 4-4 Performance graph of SSDetect for JPEG images	43
Figure 4-5 Performance graph of SSDetect for PNG images	43
Figure 4-6 Performance graph of SSDetect for TIFF images	44
Figure 4-7 Screenshot of New Case window	45
Figure 4-8 Analysis Window of SSDetect	46
Figure 4-9 Analysis Process Window	46
Figure 4-10 SSDetect Analysis Window - 1	47
Figure 4-11 SSDetect Analysis Window – 2	47
Figure 5-1 Password Protected Image Steganography	51
Figure 5-2 Examples of Degradation or Distortion in image	52

LIST OF TABLES

Table 1 Scan Result for modified or removed file extensions.....	38
Table 2 Performance Analysis of SSDetect for different file size images	40
Table 3 Scan result for Stego detection	41
Table 4 SSDetect Comparison	42

KEY TO ACRONYMS

DCT	Discrete Cosine Transform
LSB	Least Significant Bit
TPVD	Tri-way Pixel Value Differencing
BPCS	Bit Plan Complexity Segmentation
OS	Operating System
DSP	Digital Signal Processing
DIP	Digital Image Processing
LEA	Law Enforcement Agencies
EOF	End of File
SVD	Singular Value Decomposition
PSNR	Peak Signal to Noise Ration
MSE	Mean Square Error
PVD	Pixel Value Differencing
QIM	Quantization Index Modulation
MBNS	Multiple Based Notational System
PBS	Prediction Based Steganography
HVS	Human Visual System
DROID	Digital Record Object Identification
DD DT DWT	Double Density Dual Tree DWT
BMP	BitMaP
TIFF	Tagged Image File Format
PNG	Portable Network Graphics
EXIF	Extended File Information
GUI	Graphical User Interface
IDE	Integrated Development Environment

INTRODUCTION

1.1 Introduction

This chapter includes background and problem statement of research work that has been undertaken in this thesis. Overview section includes the requirement of a sophisticated automatic model to identify and recover hidden data from image files.

Subsequent sections of this chapter include contemporary existing techniques used for data hiding, problem statement in the scenario of digital forensic investigators and law enforcement agencies.

A detailed section of this chapter contains deliverable and contribution in the context of problem statement.

The projected solution will help to identify and recover the hidden secret message concealed in image files, the proposed model will give assistance to Digital Forensic Investigator who relies on multiple tools. This model and developed software will also minimize the efforts to identify modified or removed file extension in minimum time.

1.2 Overview

Computer and Internet are biggest source of creating and transferring digital data such as text documents, images, videos and application etc. Millions of documents are created, transferred and archived every day. Computation power of devices and storage capacity is growing rapidly. The techniques of digital communication are also improving with time, now we transfer thousands of documents from one device to other device within friction of time. Beside numerous advantages of technologies it also made several overheads. As the reliability on computer technology is increasing rapidly, so the techniques to secure digital data are also improving. In the interim deceiving and spoofing techniques for user are also sophisticating with the passage of time.

Several terminologies and technologies have been invented for securing classified information. Cryptography is the most commonly used terminology to protect the critical information. On the other hand, bad guys and terrorist have also made new techniques to deceive law enforcement agencies, police and government organization. Hackers are also continuously improving their methodologies to bypass the security. One of the goals of hackers and anti-state agents is to communicate covertly and stealthily by using usual communication technologies.

1.3 History of data hiding techniques

Data hiding or covert communication has been in practice thousand years before, however with the passage of time the techniques are renewed. In ancient times the techniques for secret message transfer were different, as the messages were conveyed in written or oral form. With the evolvement of Information Technology our techniques for communication and message propagation have been transformed accordingly. The art of secret communication has also been evolved. Three modern methods for digital data hiding are Steganography, Cryptography and Watermarking. All the three terminologies are interlinked. Detailed definition of each term will be in the explained in forthcoming paragraphs.

1.3.1 Data Hiding - Steganography

Steganography is the art of embedding secret message covertly in a multimedia carrier. Multimedia carrier may be Image, Audio, Video or Text file. Steganography has gained importance due to the massive growth of internet users and secret communication over non-secure communication channels. Steganography could also be defined as the study of invisible statement that hides the existence of the secret message. Despite of illicit usage Steganography is also used for military communication, authentication and integrity of message. Steganalysis is the process or technique of detecting and revealing the embedded secret message in cover media.

1.3.2 Data Hiding – Cryptography

Cryptography is the art of storing or transmitting digital data in a way that it can't be understand by unauthorized identity. The main objective of cryptography is to attain confidentiality, integrity, Non-reputation and authentication. By applying cryptography code, plain (understandable) information is converted into cipher (non-

understandable) information. The study of decryption or analysis of code from encrypted information is called Cryptanalysis. The authorized user uses the recovery key to decrypt the information from cipher text to plain text at the other side. The encryption algorithm, key size and channel of communication are the important parameters in subject of cryptography.

1.3.3 Data Hiding - Watermarking

Watermarking is also data embedding technique, but the embedded message can only be recovered by the owner of multimedia file to recognize the identity of digital content. The main objective of the watermarking is authenticity and confidentiality of object. Watermarking does not provide integrity and non-reputation. As these three data hiding techniques have several similarities, but they can be better understood with following comparison.

1. Carrier for all three techniques be any multimedia file (Image, Audio or Video)
2. Security Key to recover the embedded message for Steganography and Watermarking is optional while in Encryption the security key is mandatory/necessary.
3. Objective or application of Steganography is to communicate secretly, Watermarking is used for Copyright preservation and Encryption is used for data protection.
4. Output of Steganography is Stego-file, Watermarked-file in case of Watermarking and Ciphertext in case of Encryption.
5. Strength of Stego file (Output file after Steganography implementation) can be tested by its Delectability / Capacity, while in Watermarking and Encryption the Strength can be verified by it Robustness.

Complete hierarchy of data hiding techniques in the subject of information security and their extension are given in Fig 1-1 shown below.

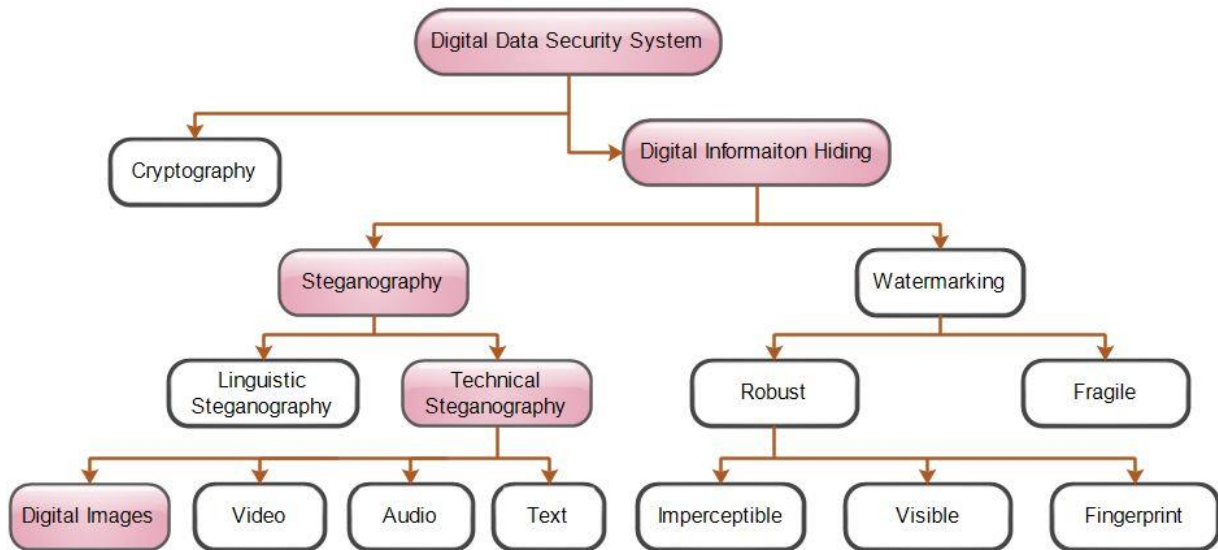


Figure 1-1 Data Hiding Techniques

The work in this thesis will be focus on Technical Steganography and digital images as carrier only. The research also emphasis on file extension hiding (file extension in Microsoft Windows Operating System (OS)) and their recovery techniques.

1.4 Application of Image Steganography

The main goal of steganography is to communicate securely in completely undetectable manner. Steganography is primarily used to carry out secret communication while maintaining confidentiality to ensure privacy. Other application of steganography is to hide individual details in photographs. TCP/IP packets (for a session ID can be added into an image to analyze the network traffic of particular users) [1], and also checksum embedding [2]. In medical steganography is used to hide patient particular i.e. Patient's name, address and ID inpatient medical reports (Medical Images) [3] . In future Steganography, can replace barcodes, where user's or owner's data could be hide in doctor's prescriptions, billboards, business cards and pamphlet's [4]. For Military or Law-Enforcement-Agency (LEA) Integrity of information is most important thing, Steganography could help in this situation also. A technique proposed in [5] [6] for a scan document, integrity can be validated by using the hidden information.

Besides these beneficiary applications, Image steganography is also used by bad guys and illegitimate persons for erroneous application. Computer Hackers use image steganography to hide malicious code behind a harmless image, which can straightforwardly deceive user and anti-virus. Stegnosplit is the next generation malware and

viruses. Stegnosplit is a complete package with malicious code and its unpacking executable in an innocent looking image. The single file could be saved as a valid HTML or JavaScript file which displayed like a single image file [7] [8]. An example of actual image and Stegnosplit file side by side with its hex code using hex editor tool are shown in **Error! Reference source not found.** “Example of Image Stegnosplit” at page 5 .

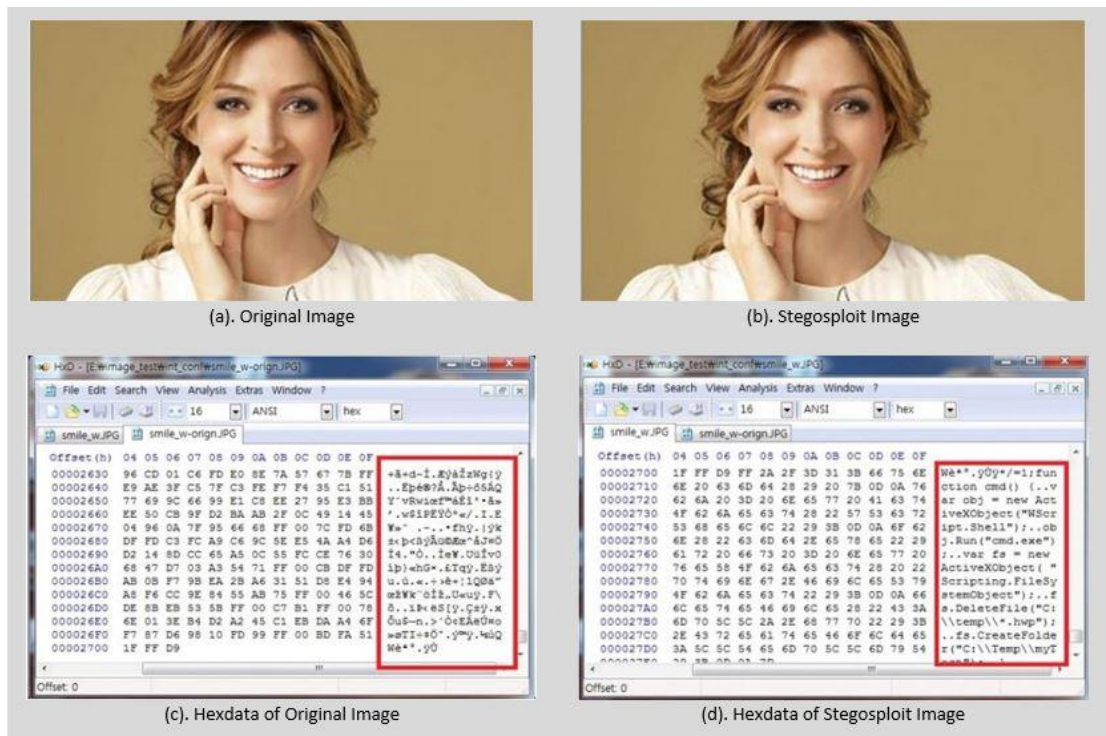


Figure 1-2 Example of Image Stegnosplit

Criminals also use image steganography to transfer their secret messages and plans covertly for intra-group communication. It is revealed after forensic analysis of computer recovered from the compound of Bin Ladin that his terrorist group use steganography for transferring their secret message [9]. This technique could be used to save important text, plans and information at local system.

1.5 Problem Statement

Steganography refers to the science of “invisible” communication which basically strives to hide the presence of message from an observer. Though steganography can be implemented in image / graphics, text, audio and video formats etc. Present techniques of steganography become more complex and even difficult to detect. This research will focus on image Steganography, which is the most commonly used tech-

nique in which secret text / image is concealed behind a benign image because of the inherent redundancies and the inability of the human eye to detect small changes in the pixel intensity values [10]. Steganography is also used by hackers to send viruses and Trojans to compromise machines, terrorists and other organizations that rely on covert operations to communicate secretly and safely [11].

Now hundreds of free and easy to use steganography programs available on internet that allow users to concealed text message in various cover media. The main aim of this work is to design a comprehensive model of image Steganalysis that detect and recover hidden message from any (or try to recover from maximum available) steganography program.

Moreover, malicious users often try to spread computer viruses and worms by changing file names with known extensions. Since the extensions are hidden by default in Windows Explorer, bad guys may send files like LOVE-LETTER-FOR-YOU.TXT.vbs to mislead common users.

Data hiding either by changing the file extension or Steganography have become favorite topic for criminals and terrorist groups, as these methods are easy to use. When a secret message is wrapped with a benign image it could be posted on public website or transferred on non-secure channel. Data hiding has also turned into a challenge for digital forensic investigator and LEA. A large improvement is required in this field to develop appropriate application and techniques that can help forensic investigator.

With the increase of storage capacity of personal computers and handheld devices, each device averagely contains thousands of pictures and files. When these devices come from apprehended suspects for Forensic Investigation, it becomes very difficult for the investigator to check each image to detect steganography. It is also very difficult to identify files for which the actual extensions have been removed or altered from mass number of files. One way is to compare the file signature via any Hex editor software but this task is manual and time-consuming.

1.6 Advantageous stockholders of research

Prime beneficiaries of this research work are digital Forensics companies or law enforcement agencies who normally recover suspicious files containing hidden information. This project can also be beneficial for Government Agencies and Military as terrorists / criminals are now well aware of current technologies and techniques. The quotation of David Clarke from the book *“Technology and Terrorism”* well describes the importance of this project.

“The United States military also maintains a keen interest in research into new steganography software and applications and the development of new Steganalysis tools”
[12]

1.7 Proposed solution or Contribution

Image steganography is mostly used for covert message communication. Currently several sophisticated techniques are available which embed the covert message in benign image. This makes detection of covert message almost impossible in plain sight. For a digital forensic investigator, it is very common to have thousands of images stored in single digital device. It is very hard to verify each image for Steganography. Suspect / attackers change the original file extension with some other extension to mislead the user or investigators. When file extensions are changed, files cannot be opened by windows explorer. In some cases the investigator misses the file as he or she thinks the file may be corrupted. In this case, the investigator checks the actual file format with file signature by using any Hex editor software, which is manual and time-consuming. So, there is need for a solution that can verify a complete drive or images to isolate the Stego images and also the files for which the actual file extensions are altered or removed.

This research will endeavor to develop a Steganalysis algorithm that will be capable of detecting Stenographic images and files with altered extensions in a certain drive / folder.

The developed algorithm / model will recover secret text that are embedded with any contemporary image Stego technique, for this reason modern / current techniques are reviewed.

The main objectives of the thesis are to develop comprehensive Steganalysis algorithm that has following features and functionality: -

1. Detection of Stego images (for commonly used image formats i.e. PNG, JPEG, GIF, TIF, etc.) in a complete drive or folder regardless of technique or algorithm used for encoding
2. Extract the secret message or concealed message from the Stego images
3. Identify and separate all files for which the file extensions are altered or removed
4. Generate report after completion of scanning (total number of images scanned, numbers of images in which Steganography identified, time taken for scanning the drive / folder
5. Extract the metadata from Stego image

1.8 Conclusion

Beside numerous advantages of technologies it also made several overheads. As the reliability on computer technology is increasing rapidly, so the techniques to secure digital data are also improving. Comparison of contemporary techniques for data hiding are elucidated in this chapter. Steganography is data hiding technique besides a benign media. Research area in modern data hiding area have been nominated in this chapter, and at the end the planned solution for the problem area are also deliberated.

LITERATURE REVIEW

2.1 Introduction

Modern tools and techniques that a basic computer user embraced to hide his valuable data will be discussed in this chapter. A comprehensive literature review about classical and modern techniques of data concealing will be deliberated in this chapter. The parameters with which a Steganalysis tool will also be part of this chapter.

2.2 Literature Review

With the progression of digital technologies, digital libraries are expanding exponentially. This is the age of Information transmission, Information storage and computation. Our techniques for storing data and communication have become more advanced. Now the ordinary mail letter has been replaced with emails and SMS (Short Message Service). Wired Telephones have been replaced with wireless mobiles. At the same time, the techniques for secret message transfer have been changed. The ancient Greece uses the shove head method for secret message transfer, they write the secret message on the scalp of the shoved head of slave. Then they allow growing the hair, the secret message has been hiding, the recipient shaves the head to expose the secret message [6].

The secret message was written on paper with liquids like milk, fruit juices, onion juice and vinegar this message become invisible when dry. The secret message became visible when the paper is heated [6].

In World War II, steganography was used for transfer secret military message among fighting parties. Microdots were one of the techniques used in World War II in which a complete documents or images is reduced to the size of a dot. Microdots are then embedded in a paper and then sent to the recipient. The recipients, on the other side, enlarge the size and print the hidden information [13], [14].

Now this is era digital technology, computers are used to conceal messages. With the development in the Internet, Digital Image Processing (DIP), Digital Signal Pro-

cessing (DSP), coding theory and Information theory the ancient message concealing become transformed into digital Steganography. Secret messages and malicious codes are concealed in pictures, music and video files.

2.3 Computer File Extension – Microsoft Windows Operating System

This research and proposed solution / methodology are focused on windows OS. Windows OS has more the 80% market share in desktop OS worldwide [15]. File extension (external signature) are displayed in windows OS (Windows 10 Professional) from the menu bar -> View -> Option -> Folder option -> View -> Hide extension for known file types (Uncheck the option).

File extension provides assistance to Operating System in determining the appropriate software program to open it. Windows OS only relies on the file extension to browse a file / document. For example, an MS word document is created with title test.doc. Windows tries to open the document in MS Word, Notepad or WordPad. When we change the extension of the file to .jpg, then Windows tries to open the document in an image viewer or other image editing application. As the file extension can be easily altered with just a few storks of the keyboard. But the removal of file extension or altering file extension can also be used for the offensive purpose. This action also makes the digital forensic investigation difficult. If an investigator is investigating a few tens of documents, he / she can check it manually, but normally there are hundreds of thousands of files in a single digital storing media. If altered file is not properly identified, important evidence may be excluded. There are few drawbacks of file extension listed below (this literature / research is specific to Windows operating system).

1. Windows OS takes decision only with file extension, these file extensions can easily be spoofed, altered or removed
2. File extension can be easily removed or changed by any user regardless of privilege and authorities
3. Different versions of same file formats mostly have same extension
4. File extensions are not standardized or unique, unrelated file formats can have the same file extension

File signature (Magic Bytes) are internal signature that can be used to identify exact file type. File signature is combination of hexadecimal values of length 0-46 bytes. File signature gives information about the software type or software version e.g.

09 02 06 00 00 00 10 00 B9 04 5C 00 Microsoft .xls with excel V-2
 09 04 06 00 00 00 10 00 F6 05 5C 00 Microsoft .xls with excel V-4

Magic Bytes are not always unique for all file types, there may be same signature for different files but it is very rare. **Figure 2-1**” Magic Word” shows hex view of Audio file with the extension ".wav". It is clear from the figure the bytes from 0-4 to 9-12 are the magic bytes.

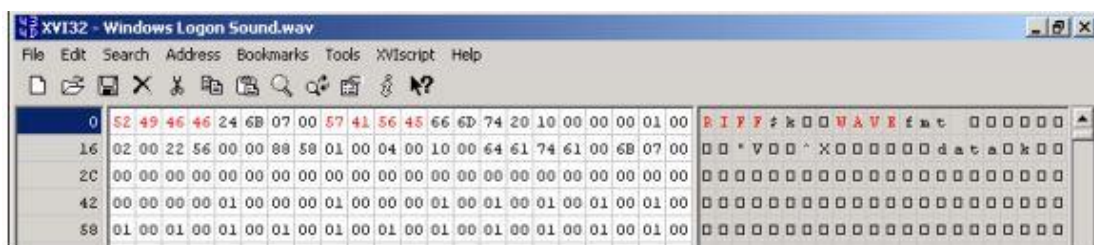


Figure 2-1 Magic Word

2.3.1 Tools and resourcing to read file signature

A few tools and techniques are available to identify actual file signature, but all the available techniques are manual.

2.3.1.1 Hex Editor Tool

Hex editor software displays the file / document in hex byte that makes up any digital object. Any Hex editor tool provides the feature to search hex values within the document. The magic bytes can be viewed and compared with the actual file signature in hex editor software. Hex software is also used to search the metadata of file, view / edit the memory partitions, data carving. Some hex editor software is 'HxD', 'WinHex', 'Hiew', 'GNU Emacs'.

2.3.1.2 Web Analytic Tool

There are few global archives that contain thousands of files signature, these global databases are frequently updated and provide free resources to search and match well-

known files magic bytes. These databases can be used to identify actual file signature. A few online file signature archives are PRONOM [16], DROID (Digital Record Object Identification) [17] etc.

2.3.1.3 Internet Browsing

Internet search engine is like "Aladdin ka Chirag" in Urdu and in English it can be said solution of everything. File signature of any file extension can be searched with few clicks on the internet.

2.4 Data Hiding Methods

Data hiding in soft form is a trendy topic, where classified data is secure in a single host or transferred from one host to another host. The most common technique used by Windows-based users is to remove the file extension or alter the file extension with another extension. Windows Operating System (OS) select the application to open the file with the file extension. When the intruder or investigator tries to open file with altered extension, he considers that the file is corrupt or damaged. Some other techniques to hide the classified data are steganography. Steganography can be implemented in Audio, Video, Image or Text format.

Commonly a steganography system consists of Cover Media and Payload (Secret message) as input. **Figure 2-2** describe complementary components in a general steganography system. The general block diagram is divided into three main blocks i.e Coding Phase, Communication Channel and Decoding Phase. Coding phase consists of Cover Image (C), a secret Message (M) and Encryption key (K). The encryption key is an optional input it is used to generate an additional security layer. These three inputs are feed into and Steganography algorithm box. The steganography algorithm gets the three inputs and generates a robust Stego image file. This Stego image could be propagated over unsecured communication medium or channel. At the decoding phase, we must have the proper decoding key and proper decoding algorithm to get the same cover image and secret message back. The recovered secret message (M hate) must be the identical as at the coding phase. General block diagram is shown in **Figure 2-2** at page 13

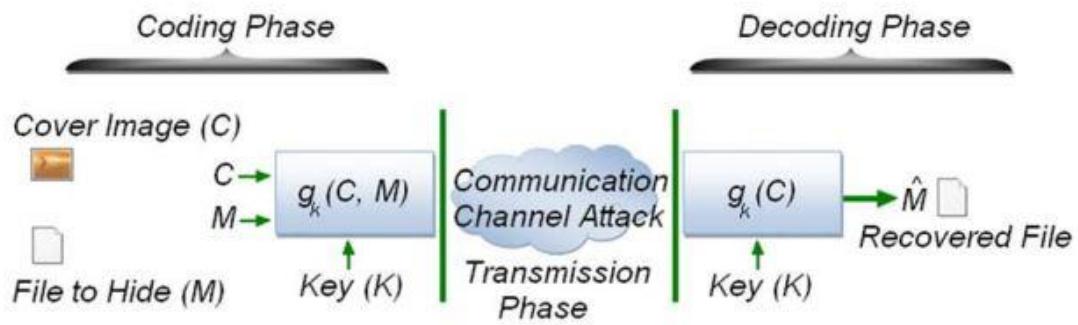


Figure 2-2 General Image Steganography Diagram

2.4.1 Steganography in Audio

In digital audio steganography, digital sound file i.e. Mp3 or WAV file used as a carrier medium. Commonly used techniques as audio steganography are LSB modified as error diffusion, phase coding and spread spectrum. Adding text message in audio file create a glitch in the sound but this glitch is normally undetectable by the human ear. [18].

2.4.2 Steganography in Video

In digital video steganography, a video file is used as a carrier. A video file is a consolidation of images and audio, due to the large size of video file, a large text file can be embedded in the video file. Video formats commonly used for steganography are MPEG, AVI, FLV, MP4, etc.

DCT (Discrete Cosine Transform) [19], LSB (Least Significant Bit) [20], TPVD (Tri-way Pixel-Value Differencing), BPCS (Bit Plan Complexity Segmentation) are the regularly used techniques [18]

2.4.3 Steganography in Images

The most desired medium used for steganography is an image, because by slightly modifying the color of images, we can embed a large number of data. This slight modification cannot be detected by visually.

In this section two most common digital image formats (Spatial and transform) we briefly discuss their uses and advantages.

The smallest unit of an image is pixel and pixel is represented by bits. If 4 bits are used to represent a pixel it will give fewer color shades, then an image which is represented with 8 bits per pixel. It is also obvious the size of a 24-bit image file will be greater than an 8-bit image file.

BMP (Bitmap), TIFF (Tagged Image File Format) and PNG (Portable Network Graphics) are types of Raster format. As these formats have the larger file size, thus larger the secret message can be embedded in these formats, but these formats are not suitable for internet as they use high internet resources.

Raster formats use pixel by pixel encoding, the JPEG (Joint Photographic Experts Group) is transformed domain format [21]. The most common used transform formats are DCT and DWT for generation of ".jpg" and ".jpeg" formats. The advantage of this image format is that they have less image size with high color resolution, but this is lossy format, which means the compressed image may have slight color change than the original image.

Before going into the explanation of sophisticated techniques used for hiding secret data, we will have a look on the methods of data hiding techniques that do not require any software tools.

2.4.3.1 Text file embedding into image file

By using Windows command line (DOS) a text file (.txt) having any secret message can be embedded with the cover image. The Stego file will be generated that will have the secret message at the end of file (EOF) [22]. If we have a cover image with title 'Cover-Image.jpg' and a text file with title 'copy /b Cover-Image.jpg + hidden-msg.txt + Stego-Image.jpg' hidden-msg.txt. We can embed the secret message into the cover image by typing in the command line, where 'Stego-Image' is the output steganography image. The Cover-Image and Stego-Image are shown in **Figure 2-3** side by side on page16.

It is very clear that there is no visual change in both images, it is also the histogram of both the images are also the same. The embedded secret message can be reversed if the Stego images are dropped in Notepad application. The secret message is seen in the

Figure 2-4 at page 166. As this is the simplest technique to hide data in image, but the drawback of this technique is that the size of the Stego file will be the sum of cover image size and the secret message file size, so we cannot embed a large secret message with this technique. As the large message, will create a significant change in the size of Stego file. The other limitation of this technique is that it is not resistant to any distortion attack i.e. cropping, re-sizing and editing etc.

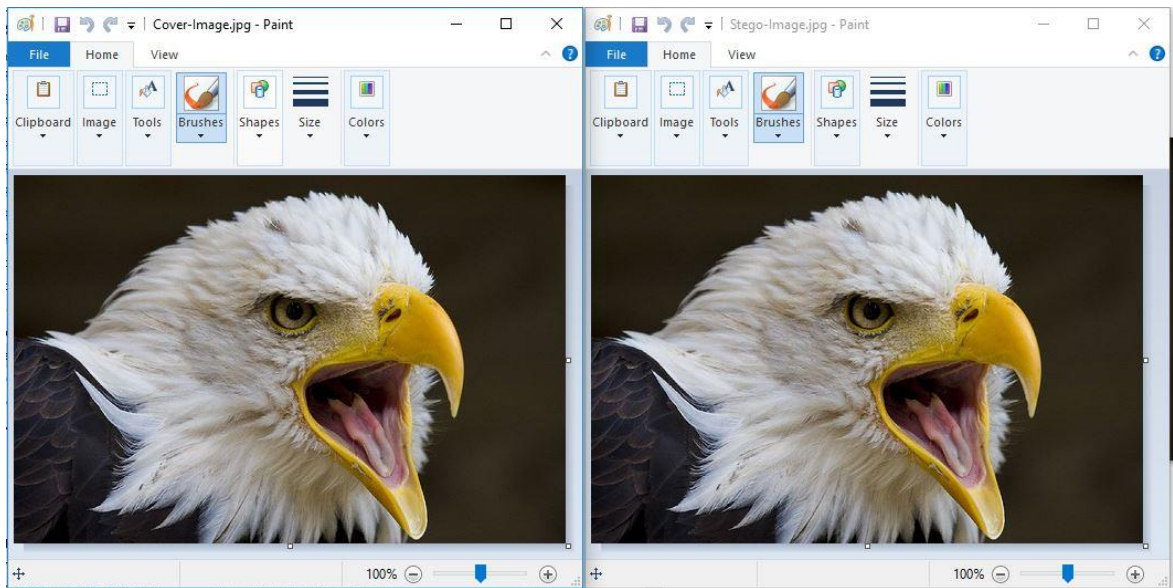


Figure 2-3 Text File Injection

Digital image creation and representation is a huge subject, but in this small Images formats such as JPEG, BMP, GIF, PNG etc are used as a carrier for steganography.

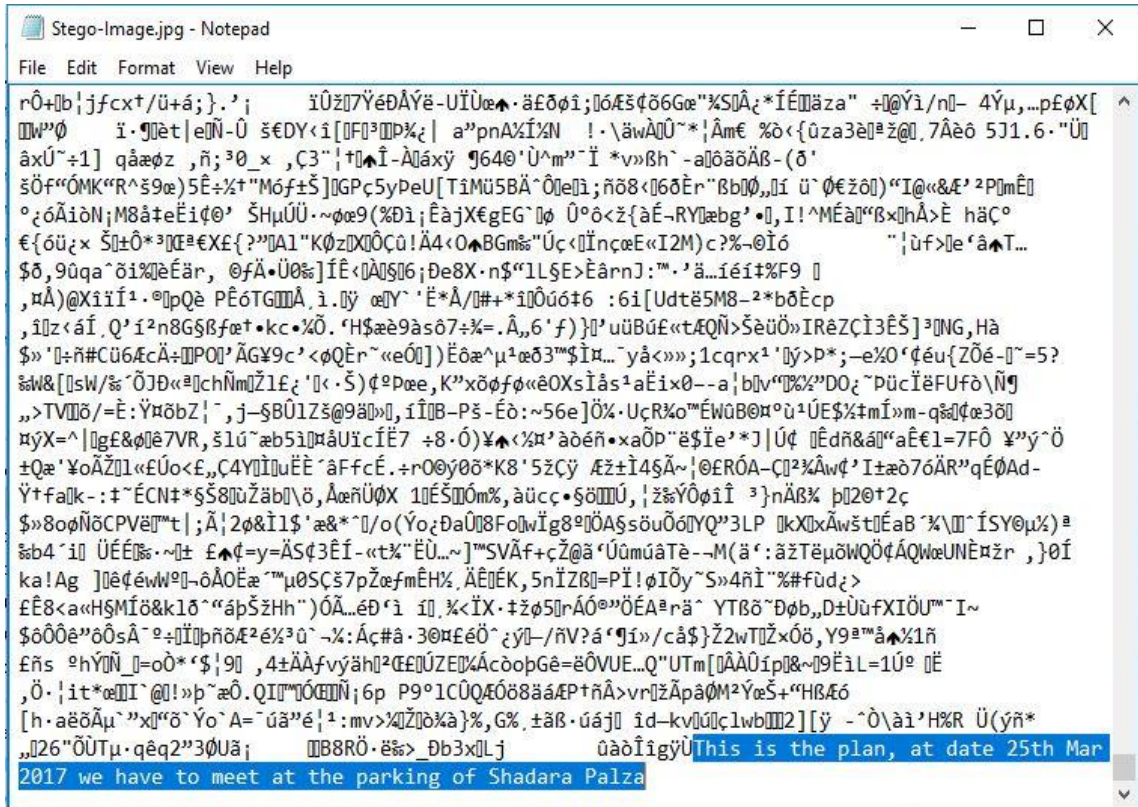


Figure 2-4 Text File Extraction

2.4.3.2 Zip / Rar file embedding into image file

We can also embed a Zip / Rar file as a secret file with an innocent image file with the same DOS command. Zip / Rar file can be a blend of multiple files. With this technique, we can have sent image files covertly to the intended receiver.

By using windows command shell, we must hide a Zip / Rar file e.g. NUSTLOGO.rar in an image file e.g. NUSTLOGO.jpg. The resultant image file is saved with title Stegno.jpg. We will use the command 'copy /b NUSTLOGO.jpg + NUSTLOGO.rar Stegno.jpg'. The size of the resultant file will have been almost the sum of cover image and secret .Rar file. The embedded .rar file can be recovered if we open the resultant image file using any Rar / Zip software [22].

2.4.3.3 Data hiding in EXIF

EXIF (Extended File Information) of an image store the metadata information (data about the data). We can easily hide our secret message in EXIF portion of an image, it can be done by right-clicking the image and choosing the properties option shown in **Figure 2-5** at page 17. The secret message can be written in the command column

comes under details tab [22]. As mostly the EXIF portion of an image are ignored. To retrieve the hidden message, the receiver just has to follow the same steps at the receiver end.

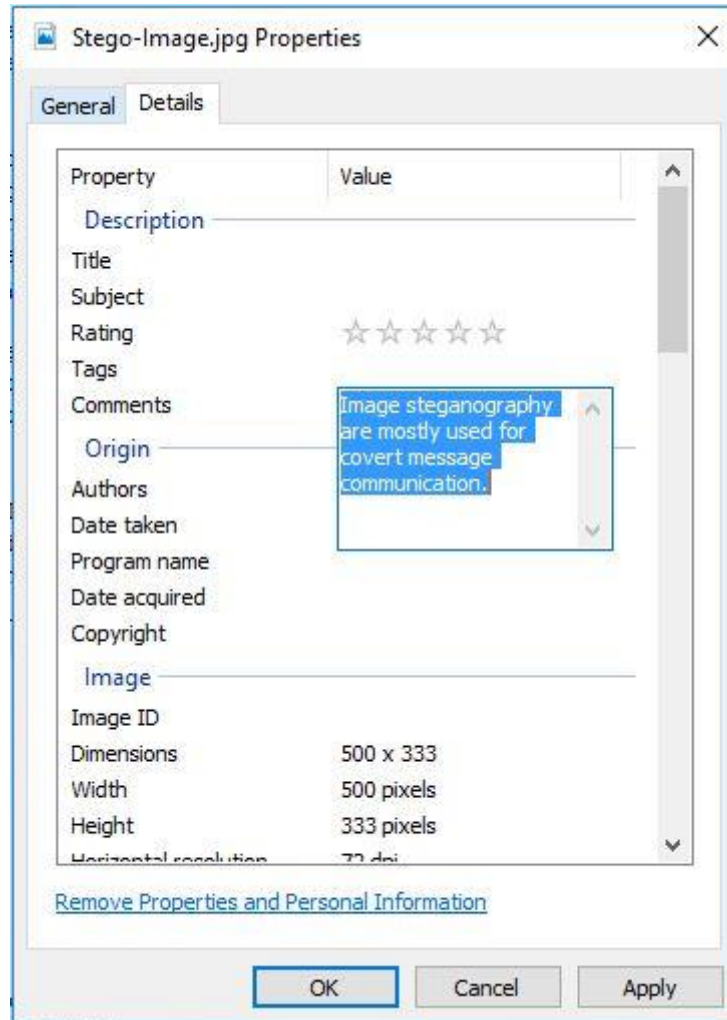


Figure 2-5 Exif Data Hiding

2.5 Image Steganography – Fidelity Criteria

In last decade, several steganography schemes are proposed, but evaluation parameters are the same. The evaluation parameters are Robustness, Imperceptibility, Capacity and Computational Cost.

2.5.1 Robustness

Robustness demonstrates the quality of steganography technique. This parameter is used to evaluate, will the final output sustain after different noise and distortion attacks.

Robustness of technique is evaluated with Peak-Signal-Noise-Ratio (PSNR) and Mean Square Error (MSE). It is measured in decibels (dB). PSNR is calculated using

$$PSNR = 10. \log \left(\frac{MAX^2}{MSE} \right)$$

where 'MAX' is Maximum Possible pixel in carrier file.

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i, j) - K(i, j)]^2$$

where 'm' and 'n' are the numbers of horizontal and vertical pixels in the case of image as a carrier.

2.5.2 Imperceptibility – Perceptual Transparency

Imperceptibility is another important parameter which should be maintained while embedding the secret message. Imperceptibility of stego image is correlated with Capacity (Payload). High the capacity less will be the imperceptibility for Stego-file.

2.5.3 Capacity

Capacity or Payload is the amount of data that can be embedded into the cover medium without the degrading its quality. Capacity is measured in terms of bits per pixel.

$$Capacity = \frac{Numberofbitsusedtohidedata * 100\%}{Totalnumberofbitsinimage}$$

2.5.4 Security

Security is the ability of an unauthorized individual to discover the hidden message. More secure the algorithm; it will be more difficult to recover the embedded message.

2.6 Computational Cost

Quality of algorithm can also be measured with the Computational cost. A technique is said finest if it requires less computational power.

2.7 Image Steganography Techniques

Image steganography is usually divided into categorized into four types i.e. spread spectrum, spatial domain, transform domain and model-based steganography as de-

pictured in **Figure 2-6** at page 19. Details of each domain will be in the coming paragraph.

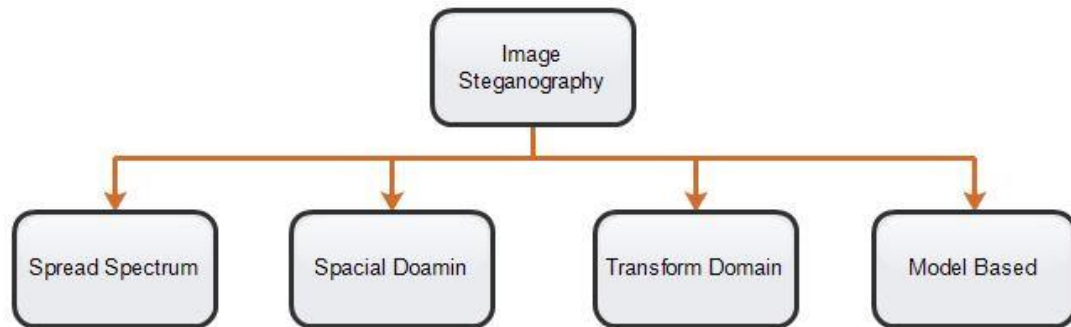


Figure 2-6 Image Steganography Techniques

2.8 Spacial Domain - Image Steganography

In Spacial domain, the secret message is directly embedded into pixel value. Most common approaches in this domain are listed given in **Figure 2-7** at Page 19.

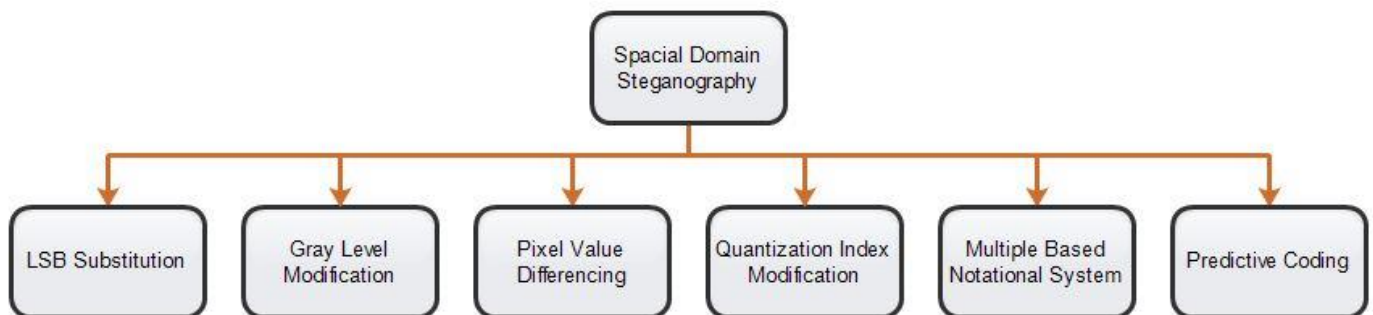


Figure 2-7 Image Steganography in Spatial Domain

Least Significant Bit (LSB) techniques least bit of the cover image is replaced with the secret message. This is the most commonly used technique, as this gives high imperceptibility and simplicity. Due to its simple implementation, many variations are proposed to improve the algorithm and its robustness. A few recent and improved algorithms are Dual layered secure algorithm [23], Block-based LSB algorithm [24] and block level entropy algorithm [25]. Drawbacks of this technique are it is not very robust, it does not sustain for small distortion and noise attack.

Gray Level Modification was first time presented by Potdar et al in 2004 [26] . In this method, the gray levels of cover images are modified by the secret message. This technique gives a high capacity of less computational complexity [27].

In **Pixel Value Differencing (PVD)** technique [28] the cover image is divided into non-overlapping blocks, then arranged with the difference between each pixel. The secret message is embedded into the pixels where the difference is large. Large data is embedded at the edges in the picture where there is a sharp change of color. Advantages of this technique include high capacity with high imperceptibility [29].

In **Quantization index modulation (QIM)** cover medium is first modulated with an index or sequence of indices with the secret message and then quantized with the associated quantizer or sequence of quantizers [30].

The carrier medium is converted into series of symbols with different base system other the binary base system. In **Multiple Based Notational System (MBNS)** [31] More information is embedded if the greater base system is used.

Prediction based steganography (PBS) Embedding by altering the pixel values directly leads to significant distortion in Stego image resulting in less hiding capacity and poor visual quality. To overcome this issue, the predictive coding approach is suggested where pixel values are predicted using predictor and instead of altering the pixel values, prediction **Error Values (EV)** are modified to embed secret data [32].

2.9 Transform Domain Image Steganography

Cover images are transformed into frequency domains, which is a combination of low and high-frequency components. Smooth part of images creates low frequencies while the edges and sharp parts of image create high frequencies. High frequencies can sustain for large variation; we can embed large input data in this part of images. This transformation into frequencies gives advantages to exploit **Human Visual System (HIV)**.

The cover image is first converted from spatial domain into the frequency domain by using any forward transform technique, the secret data are embedded by altering these transform coefficients. At the end inverse of the transform is applied to construct the cover image. The generic block diagram for transform domain image steganography is given in **Figure 2-8** at Page 21. A few transform used for image steganography are DCT [33], DWT [34], Ripplet transform, Hadamard transforms [35], Ridgelet transform, Haar transform, DD DT DWT, etc. The implementation of these transform is same as at the encoding side an appropriate type of transform and transform coefficient is selected depending on the nature of the cover image.

These transform coefficients are altered to hide secret data. With the help of desired embedding algorithm, secret data can be embedded in suitable transform coefficients. Now, apply inverse transform to derive Stego image. For extraction, similar steps in reverse order are performed to recover cover image and secret data.

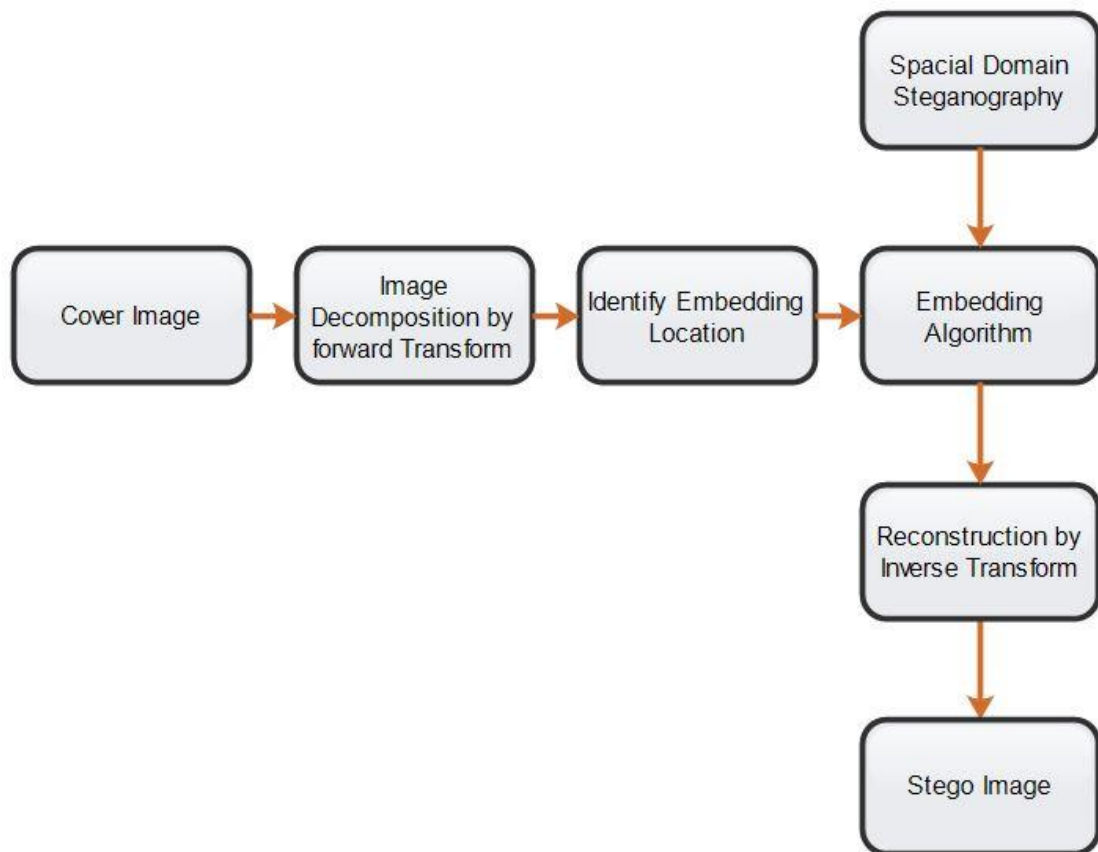


Figure 2-8 Transform Domain Image Steganography

In Spread Spectrum Image steganography, the message data is divided into patches and each patch is spread across the frequency spectrum of the carrier.

2.10 Steganalysis – Steganography Detection Techniques

The method or technique used to recover embedded / planted message in digital media (Image, Audio or Video) is called Steganalysis. The history of Steganalysis is as old as steganography. As the techniques and methodologies for embedding confidential information in media are transforming, at the same time recovery or decoding techniques are also renovating. These new technologies produce opportunities to the hostile adversaries to communicate covertly. The responsibilities of LEA to protect their classified information from intimidating culprits have become more and more challenging. Knowledge of Steganalysis is benefited in the domain of digital Forensics, tracking the criminal activities over the internet. Steganalysis techniques cover in this research paper are not ultimate, these methodologies will be evolved with the advancement of new steganography technique.

In this research the most promising techniques for Steganalysis are identified. Broadly these techniques are classified into two fields, signature Steganalysis techniques, and statistical Steganalysis techniques. Each technique is further divided into explicit (specific steganography method) and universal approach.

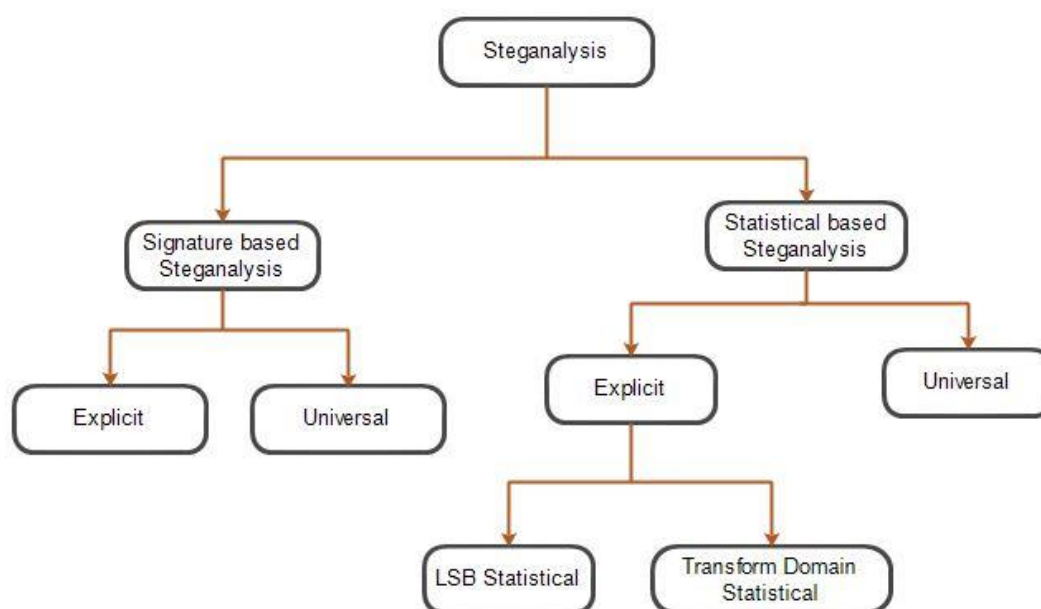


Figure 2-9 Steganalysis Techniques

The complete hierarchy and grouping are given in **Figure 2-9**. Further details of each technique will be covered in coming section.

2.11 Cataloguing of Steganalysis techniques

A complete hierarchy of Steganalysis techniques is given in **Figure 2-9**. Steganalysis techniques are broadly divided into two domains, signature-based steganography detection and statistical based steganography detection. Each domain is further divided into explicit and universal approach.

2.11.1 Signature-based Steganography detection

The objective of the steganography is to hide some secret information into digital media without degrading its imperceptibility, while embedding this additional bit of information into digital media it modifies some characteristics and leaves some specific pattern into the media. These specific pattern or characteristics are called signature. These signatures give suspicion of the existence of the embedded message. So in signature-based detection the Steganalysis tool gaze for the specific pattern in the media to expose the probability of hidden message.

2.11.1.1 Explicit Signature-based detection

Many steganography tools left some specific pattern or signature while embedding a secret message into the cover media. These hidden signatures help to retrieve the coded information speedily [36].

Hide and Seek is an image steganography tool that left its signature in the Stego-image [37]. A different version of this tool left a different signature, these specific signatures can be verified and retrieved by comparing the actual cover image with the Stego-image in hex editing tool.

Jpegx is well-known image steganography software. It is investigated that this tool inserts a fixed signature before adding the secret message. This tool always adds the secret message at the end of the JPEG file. This signature can also be verified with the hex viewing software. The explicit signatures in Hex form are: 5B 3B 31 53 00 [38]. By identifying these specific patterns in any image implies the presence of hidden information.

2.11.1.2 Universal signature-based detection

Universal signature based steganography detection is proposed by [39] [40] [41]. In his study, he discourages the use of JPEG format as the cover image for steganography as JPEG compression introduced a unique fingerprint in the form of quantization. If secret information is embedded in JPEG image in the spatial domain, this steganography image can easily be identified by inspecting its quantization matrix with the actual neat JPEG quantization matrix. In this technique, the Stego image is first divided into 8x8 block, then quantization matrix is created by DCT coefficient in all 8x8 blocks. This quantization matrix is then compared with the quantization matrix of standard JPEG quantization table. This universal signature-based detection is very reliable to identify the Stego-image.

2.11.2 Statistical based Steganography detection

The smallest unit of a digital image is pixel and pixel is represented as binary / numerical value in computer. If we embed some additional information in an image it employs some change in statistics of image value. In statistical Steganalysis, we perceive this deviation or alteration in the image pixel values.

2.11.2.1 Explicit statistical based detection

In explicit statistical based steganography detection technique, we notice a specific variation in image property due to definite encoding technique.

- ***LSB statistical Steganalysis***

LSB (Least Significant Bit embedding) is the mostly used steganography method due to its simplicity. In LSB technique the secret message is embedded in the least significant bit value of the cover image. Numerous free steganography tools available on internet use LSB embedding technique [42].

Fridrich et al proposed the first statistical Steganalysis technique for a color image in Aug 2000 [43]. Fridrich uses Raw Quick Pair (RQP), he analyzed that close pairs of colors increase significantly to the ratio of a total number of unique color in an image after embedding the secret message. The drawback of this technique as it gives better detection of the color image then the grayscale images.

Avcibas et al in September 2002, proposed another statistical technique to detect LSB steganography [44]. His technique was simple and efficient, in which he gets the 7th and 8th bits of the Stego image and find the correlation of these two bits' value with the other bit planes that are affected encoding the message [45].

- ***Transform domain Statistical Steganalysis***

Detection of steganography in wavelet domains is proposed by S. Liu et al in 2004. He proposed that histogram of a clean image is smoother than Stego image. He suggested spectrum analysis and energy differences in a histogram to detect the clean and Stego image [46].

2.11.2.2 Universal statistical based detection

Universal statistical detection uses to detect the Stego regardless of embedding technique. Universal detection is the hardest methods in the terms of computational power and encoding, they use Neural Networks, clustering algorithms and other experimental data to detect the steganography.

2.12 Conclusion

Numerous classical and modern image steganography techniques are discussed in this chapter. The factors from which the best steganography algorithm can be evaluated are also deliberated. Steganalysis is the reverse technique to recover the embedded message. Specific and blind attacks are the two common types that are employed to recover the encoded messages. In this chapter a complete review of all possible attack to recover the encoded information are covered.

DESIGN AND METHODOLOGY

3.1 Introduction

This chapter will explicate the proposed approach to develop an automated and hybrid model that will undeniably minimize the effort of Digital Forensic investigator, to recover the wrapped text in image file. The framework adopts to develop the software tool, different tabs with their functionality, software user guide or tutorial will also be explained in this chapter.

3.2 Design Goal

With the increase in computational strength of personal computers and digital portable gadgets, the creation of digital data is rapidly increased. Also, the techniques / methodologies adopt to hide secret data by hackers / intruders / terrorist from the LEA are also modified. Data hiding techniques are used to hide and secure personal or private information to save over a local system or to send other users. Very specific to steganography and data hiding the techniques are also rapidly evolved. Undeniably cryptography gives best results to secure the important data but it also alerts the intruder that data contains some important information, while steganography has the advantage that data is hidden behind benign files. Sophisticated techniques cannot be detected by human eye.

The prime focus of this research is to create a universal Steganalysis methodology that decodes any steganography image regardless of its embedding techniques. This research is to create a hybrid model that detects the embedded secret message with signature-based detection and statistical based detection algorithm. The automatic software will also detect the files for which their file extensions are altered or removed.

3.3 Existing similar programs and their limitations

As there are many free software programs available on the internet for image steganography, these stego images can only be decoded with the same tool with which the message is embedded. Every author or programmer tries to write a robust and imperceptible algorithm. As already mentioned in chapter 2, there basic two types of possible attacks to recover the hidden information from stego files. There is no such hybrid program available that can brute force these two techniques on a single drive to the directory. Three common and mostly used algorithms for image steganography are VSL (Virtual Steganography Library), StegDetect and StegSecret. A brief detail of each algorithm, their strength and limitation is discussed in coming sections.

3.3.1 VSL (Virtual Steganography Library)

VSL is written and maintained by Michał Węgrzyn and available at <http://stegsecret.sourceforge.net/#downloads>, the source code is freely available in Java language. VSL is Steganographic and Steganalysis tool. LSB algorithm with advanced Karhunen-Loeve Transform (KLT) technique and DCT are used for image steganography. This algorithm provides robust embedding in JPEG images. For Steganalysis this algorithm uses Binary Similarity Measures (BSM) method with Support Vector Machines (SVMs) classifier and RS-Analysis algorithm (Regular groups and Singular groups) [47].

Even though this is the best algorithm for Steganalysis but the algorithm results are satisfactory only in LSB embedded images, the Steganalysis scope is also limited for Jpeg image format. This algorithm has imperfect results in case of blind Steganalysis (universal) technique.

3.3.2 StegDetect

StegDetect is command line Steganalysis algorithm that uses statistical based LSB stego detection. This algorithm is best for JPEG image format; it also gives limited detection of steganography in spatial domain. This algorithm can detect the presence of JSteg, JPHide, OutGuess and Camouflage. [48]

This is the best algorithm for detection of secret message that is embedded at End-of-File (EOF) steganography. Complete code of StegDetect is available at <http://www.outguess.org/detection.php>

3.4 F-5 Steganography Algorithm

F-5 Steganography Algorithm is written and maintained by Andreas Westfield, this algorithm becomes stable after several advancements, [49]. The initial versions of F-5 are JSteg, F-3 and F-4. This algorithm is the most commonly used image encoding and implemented in free stego programs. The preliminary versions hold some limitations i.e they were the weak for virtual and statistical attacks. The preliminary versions also contain restrictions for low capacity input secret messages.

3.5 Hybrid Steganalysis program

The aim of this project is to detect image steganography implemented with any stego algorithm and recovers the encoded message. The designed hybrid model will recover the image with signature detection and with statistical detection.

3.6 Design Features

SSDetect (Signature & Stego Detect) - provides following features.

1. Detect Steganography (Steganalysis)
 - a. User can scan for a specific file or complete drive / folder
 - b. User can select a specific technique for Steganalysis attack (to expedite the scan result)
 - c. User can apply a filter on file format for Steganalysis (to expedite the scan result)
 - d. Complete results after scanning will be displayed containing numbers of file scanned, numbers of files in which steganography is detected, recovered secret message
 - e. Provide option to the user to save the scan result

2. Detect Modified or Removed file extensions
 - a. Select system drive / folder or a file to scan for modified or removed file extension
 - b. Select system drive / folder or a file to scan for steganography and modified or removed file extensions at the same time (Slow results)
 - c. Provide option to the user to save the scan result

3. Embedding Secret Message (Steganography), this module was out from the scope of the thesis, but this module is added to create a data set, that is latterly used to evaluate the Steganalysis application.
 - a. The user can select Stego techniques, three most commonly used image Steganography technique are listed in software.
 - b. User can select image file type e.g. *.jpg, *.png, *.tif, etc
 - c. Steganography can be implemented in color images and gray-scale images
 - d. Image file information will be displayed to the user (File dimensions, file size)
 - e. User can type a short secret message in the provided text window or can load *.txt file in case of a long secret text file
 - f. Imperceptibility of both images (Cover image and Stego image) will be calculated and displayed
 - g. User can select the location of Stego image file

3.7 Design Framework

A software framework that describes complete features and implementations is given in **Figure 3-1** “SSDetect Framework” given at Page 300.

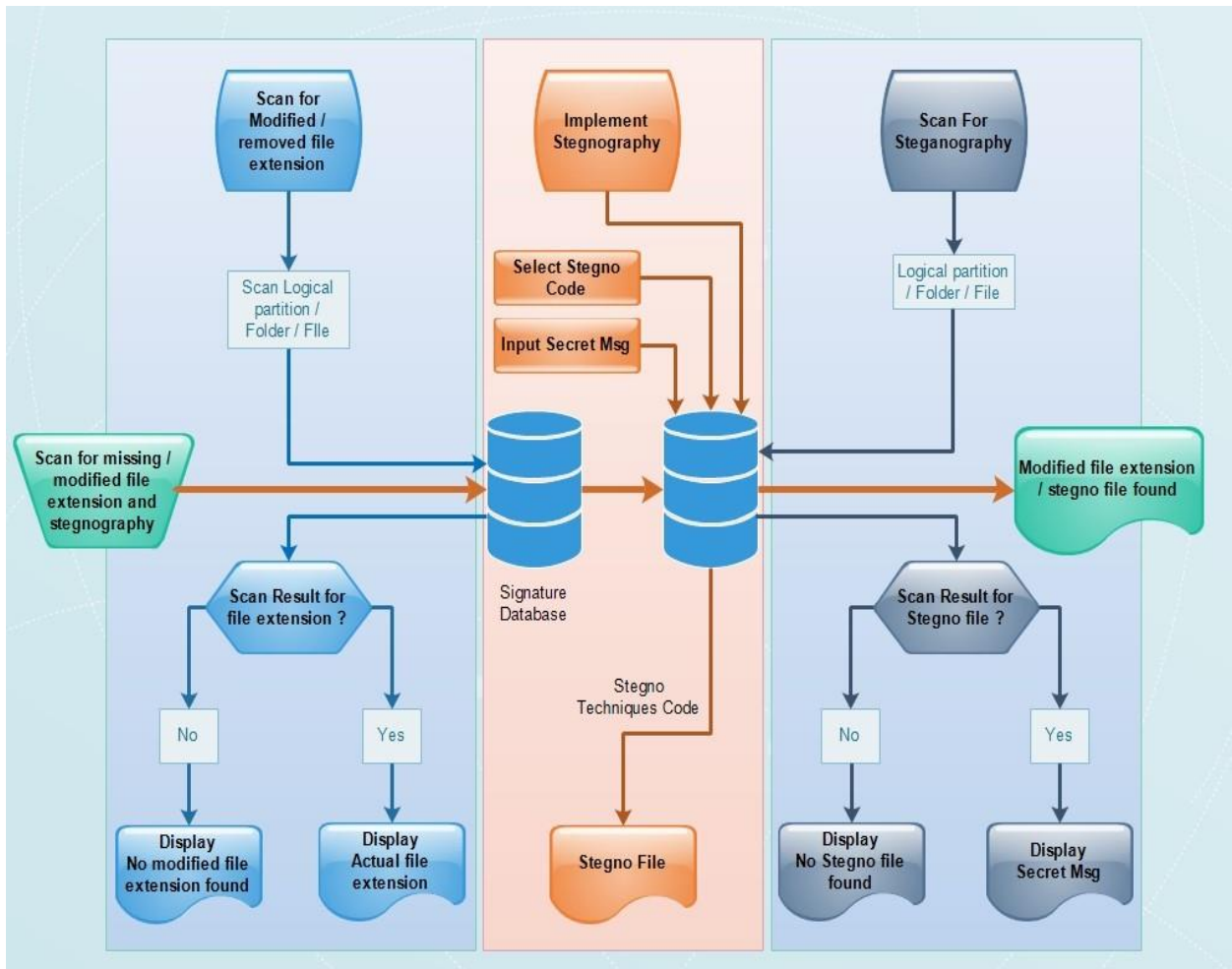


Figure 3-1 SSDetect Framework

3.8 Design Architecture / Platform

Design architecture includes the programming and scripting platform selected for software development. The entire image steganography algorithm for encoding and decoding are written in Matlab (ver R2013a). Brief details of Matlab will be given in succeeding section. While GUI (Graphical User Interface) is designed in Microsoft Visual Studio 2015 as Visual Studio C Sharp provides best plugins for GUI design.

3.8.1 MATLAB

MATLAB stands for MATrix LABoratory. MATLAB was developed by LINPACK (linear system package) and EISPACK (Eigen system package) projects.

MATLAB is a high-performance language for technical computing. MATLAB is mostly used for computation, visualization, and programming environment. Further-

more, MATLAB has become a modern programming language environment: it provided sophisticated data structures, contains built-in editing and debugging tools, and supports object-oriented programming.

3.8.2 Microsoft Visual Studio

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. Visual Studio can be used to develop console and graphical user interface applications. It can also be used to develop Windows Forms applications, web application, websites development, web applications, and web services. MS Visual Studio also supports several OS platforms i.e. Microsoft Windows, Windows Phone, Windows CE, .NET Framework, .NET Compact Framework and Microsoft Silverlight.

3.9 Design Parameters

In order to provide the complete deliverable outcome of the research, the developed software has following parameters.

3.9.1 Steganalysis - Extracting Secret Message

The prime object of this thesis is to recover / decode secret text messages embedded in graphic images. This requirement comes for the digital forensic investigator, as they

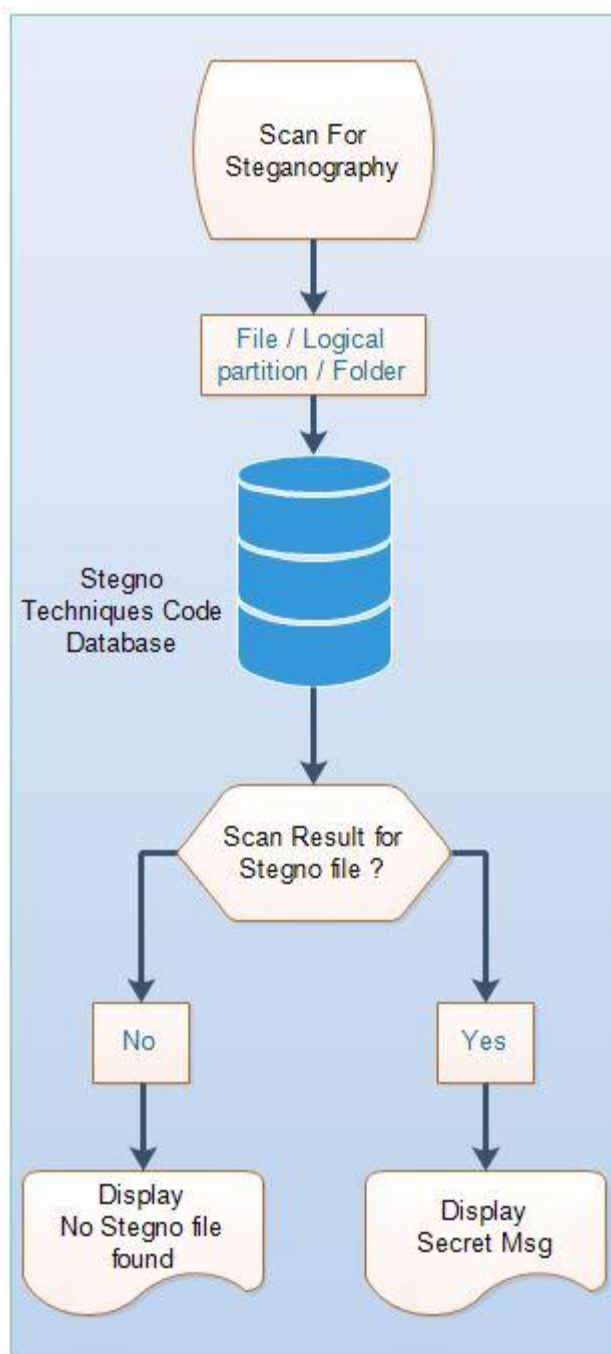


Figure 3-2 Extracting Secret Message

face several thousands of images in a single digital gadget, and it became almost impossible for them to scan each image file one by one. Another reason is if the investigator knows about the stego image he / she must find the same application to recover the hidden information with which the message is embedded. SSDetect can scan complete logical drive of system / folder / single file to discover a Stego file. SSDetect discovers the Stego file and recovers the embedded hidden message. The developed software has the advantage to scan all images in a specific drive / folder. SSDetect scan all images one by one and apply the reverse stego attack. SSDetect tool scans three common image formats for Steganalysis, but the user can select specific image format i.e. .JPG, .PNG or .TIF. Specific search i.e for specific Stego algorithm or specific image format gives faster and efficient scan.

After completing the scan for Steganalysis, SSDetect will give a scan result that contains the numbers of files have been scanned and a number of files that contains Stego message. This result can be export in text format.

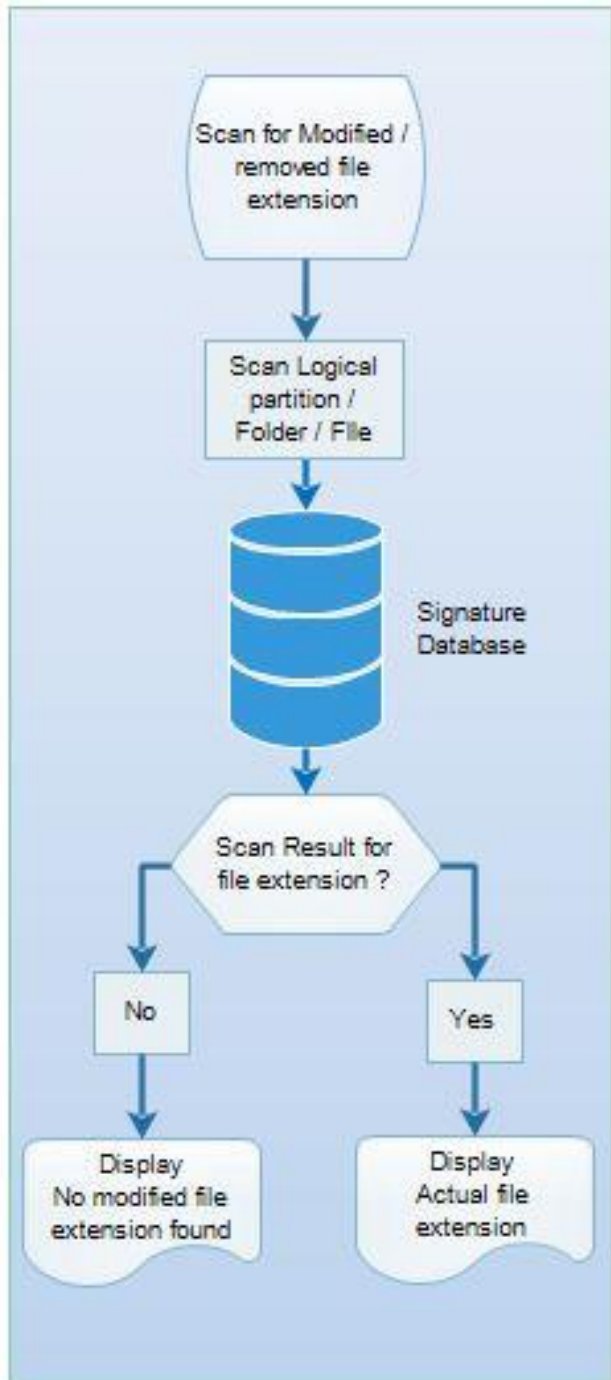


Figure 3-3 Scan for Modified or Removed File ex-tensions

3.9.2 Scan for Modified or Removed file extension

Removing or modifying file extension is also very common way used to hide important data from other users or investigator. Normally a basic computer user, who

doesn't have knowledge about other data hiding techniques, change or remove file extensions. File extensions can be modified or removed with just a few clicks, this technique can easily deceive the investigator. In Microsoft Windows Operating System an application can only understand file by its extension.

SSDetect can also scan complete logical drive / folder / file for modified or removed file extension. SSDetect matches file extensions with file signature (magic words) if it finds any change, the tool will recommend actual file extension. For this purpose, a database is developed that contains common documents signature. A list of common file types incorporated in SSDetect is given at section 3.5.3.1, however complete list with file types with their signatures are given at Appendix-III. If a user wants to incorporate a new file type, it has the option to add new file type signature. After completing the scan SSDetect generate results for all modified file extension with the path.

3.9.3 Embedding Secret Message - (Steganography)

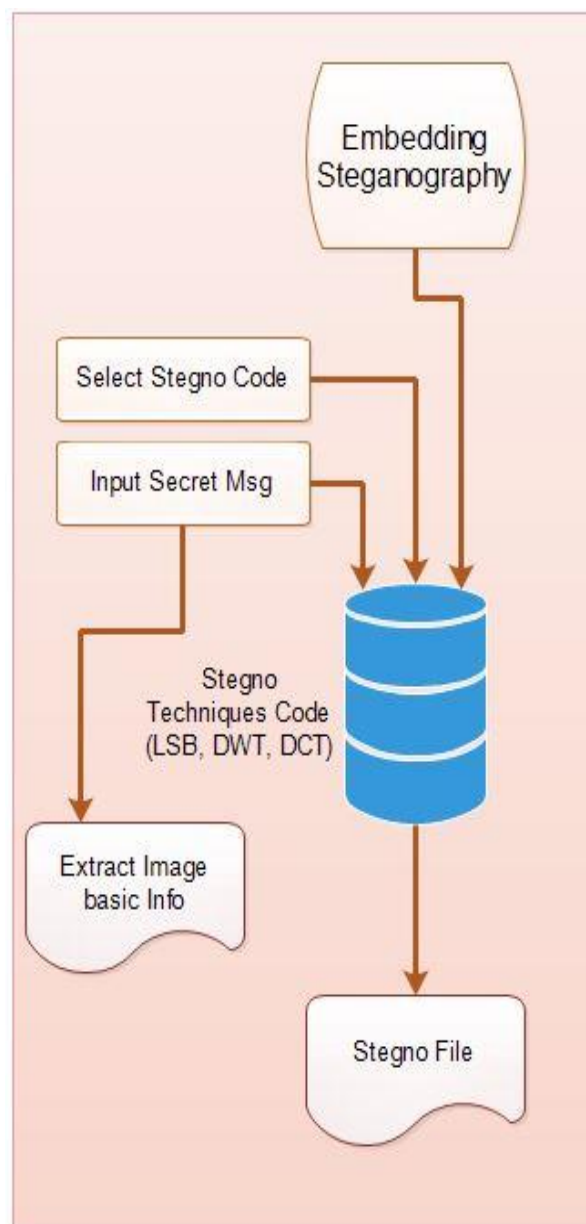


Figure 3-4 Embedding Stego Message

SSDetect gives choice to the users to select one technique, among commonly used image steganography algorithms. Most common image steganography techniques LSB, DCT and DWT are embedded in SSDetect. One technique is taken from Spatial domain (LSB) while the other two techniques belong to Transform domain (DCT and DWT). The user can also embed the secret message in three most common image formats i.e. JPG, .PNG and .TIF.

Detailed technical and general comparison of three image formats is given in Appendix-II. It is always preferred to use .PNG or .TIF as cover image as this is the lossless image compression. While .JPG is lossy image compression. These three techniques provide robust and imperceptible steganography. Png and Tiff image format can survive among several distortions and noise attack. The software is designed user-friendly, all the interface and buttons are labels with understandable titles. SSDetect also shows few details of the selected cover image i.e Display cover image, Image dimensions (LxW), Image size (Kb and Mb) Figure 4-4 at page 39. It is also referred that high-resolution image is selected as cover image, as high-resolution image has the capacity to hide a large secret message. High-resolution image has rich details or feature, after embedding the message it creates less distortion in image imperceptibility.

Secret message can be given to the SSDetect by two methods, if the secret message is short an input text windows are given, the user can write the message in the text box. If the user wants to embed a large text file to embed he / she can load the text file.

When the secret message is successfully embedded into the image file, resultant change in the Stegno image are calculated using (Mean Square Error) and displayed. A Higher value of MSE is not recommended as it shows the large change in image imperceptibility with respect to the cover image.

A complete catalog of common free steganography applications available on the internet is given at Appendix-I. All the available free / preparatory software use only one technique to embed a secret message in the cover image.

3.10 List of common file types.

Files data types are categorized as follow.

- MS Office documents
- Images
- Videos
- Audios
- PDF Documents

- Virtual Drives
- ZIP / RAR documents

The user can scan for image steganography and modified / removed file extension at the same time. In this case the user just selects system logical drive or folder, SSDetect will scan complete drive or folder for Image steganography with the available algorithm, and it will also scan each file with its given extension and file signature at the same time. But this possibility will take sufficient time, so high patience is required.

3.11 Methodology

When designing / developing a software project commonly two methodologies are adopted, Agile and Waterfall. After literature review and focuses on deliverable of this research Agile methodology are chosen for the development. Another reason for selecting the Agile methodology as it allows to incorporate new code to increase the performance of software program. Each Steganalysis code is tested independently, and afterward assimilated into the main software. The development methodology of the Agile technique is shown in **Error! Reference source not found.** at Page 35.



Figure 3-3 Agile Software Development Model

3.12 Conclusion

The objective of research work is clearly elaborated in the previous chapter, while in this chapter complete framework to achieve the objective are stated. Deliverable of the research work is divided into three modules, the outline of each module are described in detail.

IMPLEMENTATION AND TESTING

4.1 Introduction

This chapter will elucidate the implementation of SSDetect. Performance evaluation of different scenarios and testing. The functionality of SSDetect will also be explained in this chapter.

4.2 Forensic assessment / evaluation of Steganalysis tool

After strenuous research the assessment of Steganalysis tool was another challenging task. One of the problems while assessment of SSDected tool is to gain a confident belief that this software can almost detect and decode any image Steganographic. Many steganography authors and software developers keep their embedding and decoding algorithm secret, therefore creating decoding software for such embedding tool is very difficult [50]. In a practical scenario, the encoded image may have some noise and compression due to the communication medium. To create a real situation a few images with random noise and compression have also been added in test image dataset.

Digital Forensics becomes an important subject for law enforcement agencies and criminal's investigation agencies. Digital Forensics investigators get digital evidence that points to some crime, secure digital evidence, recover materials that indicate the crimes. Nowadays investigating cybercrime or copyright-related cases Steganalysis is considered an important and essential application for law enforcement agencies [51]. New techniques and algorithms for data hiding in digital media become a big challenge for digital Forensics investigators to detect existence of hidden contents through visual examination. A few automated tools are available, but these tools cannot be considered as global Steganalysis software [52].

4.3 Scan result for modified or removed file extensions

File extension modification or removal is rudimentary techniques used by the basic computer user to hide the secret files. A forensic investigator in LEA frequently notices file without file extension or modified file extension during analysis of suspect

computers or laptops. But practically, in most of the cases the investigator has time constraints; there are high chances that the investigator may skip some files, which are without the file extension. SSDetect have the ability to scan a complete drive or a folder for altered or removed file extensions. For the testing purpose, different drives with dissimilar capacity are scanned for detecting of modified file extension. Initially file signatures of 48 common file types are added in SSDetect, but the flexibility of incorporating user own file types is also embedded. A list of common file types used by SSDetect is given in chapter 3 section 3.9. Blow table gives scan results of SSDetect for various computer drives with different capacity and for a different number of altered / removed file extension. The time is taken for scan a drive also depends on system performance (System RAM and CPU performance).

Table 1 Scan Result for modified or removed file extensions

Ser	Logical drive or folder	Drive / folder capacity	Number of files scanned	Number of altered or removed file extension found	Time to complete the scan result
1.	Logical Drive	120 GB	11250	85 / 85	5 min 48 sec
2.	Logical Drive	80 GB	8254	47 / 47	4 min 25 sec
3.	Logical Drive	40 GB	1570	31 / 31	3 min 3 sec
4.	Folder	300 MB	758	25 / 25	45 sec
5.	Folder	100 MB	347	18 / 18	23 sec
6.	Folder	50 MB	98	7 / 7	14 sec

4.4 LSB Steganography detection

The performance of hybrid / universal stego detects for LSB embedded messages have been evaluated with various images of different formats downloaded from the internet. A small secret message has been embedded in these images and clearly labeled for identification. The secret messages are encoded with PySteg software. It is free and open source stego software written by “Robin David”. The performance of SSDetect program has been tested for different images with different size. For POC (Proof of Concept) the developed hybrid program is tested for limited no of images.

Outcomes of SSDetect for given set of clean and stego images fall into one of the four possibilities. For two given inputs and four possible outputs can be represented as two-by-two contingency table.

	Predicted Stego (P)	Predicted Clean (N)
Actual Stego (p)	True Positives (TP)	False Negatives (FN)
Actual Clean (n)	False Positives (FP)	True Negatives (TN)

Figure 4-1 Contingency Table

If true stego image is detected as stego image and decoded properly, it will be considered as True-Positive (TP), for true stego images is detected as the clean image will be considered as False-Negative (FN). If a clean image is detected as stego image this detection will be False-Positive (FP) and vice versa if a clean image is predicted as a clean image this will considered as True-Negative (TN). The evaluated results to detect LSB encoded stego images are between 80% to 90%.

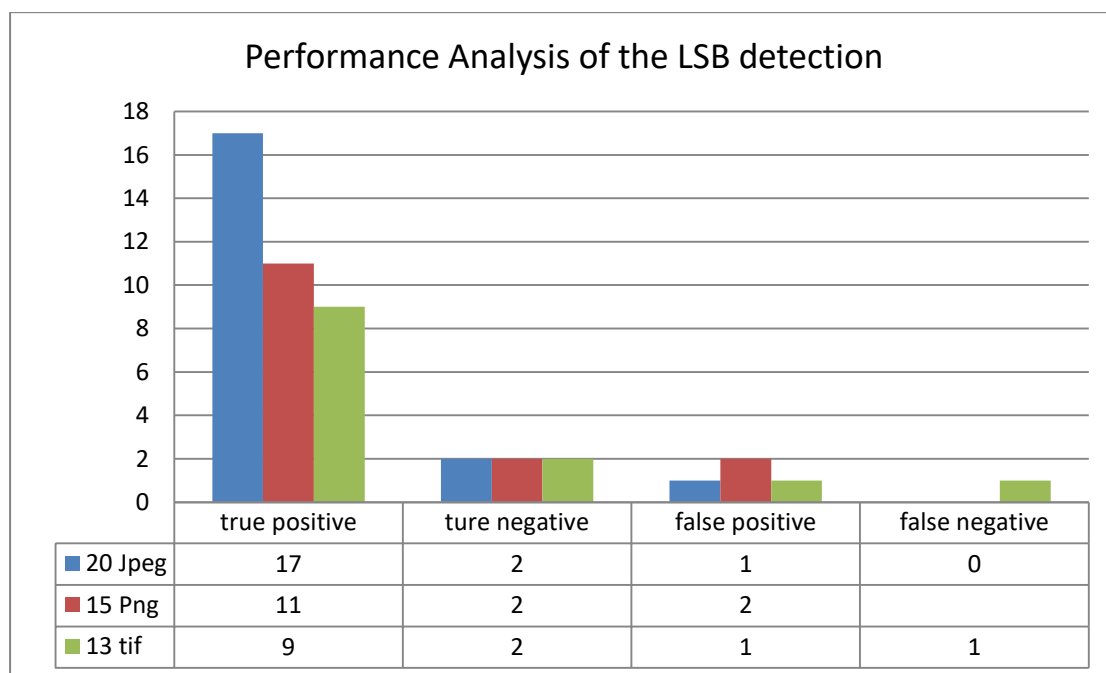


Figure 4-2 Performance Analysis of the LSB detection

Table 2 Performance Analysis of SSDetect for different file size images

Number of images	Type of files	File size	Time process seconds to in	True positive	True negative	False positive	False negative
20	Jpeg	256kb-1.27mb	10.2252	17	2	1	0
15	Png	7.98kb-524kb	8.5522	11	2	2	0
13	tif	2.54kb-752kb	7.2524	9	2	1	1

4.5 Detect Steganography with SSDetect

Most of the suspects, hackers or terrorists use image steganography to forward coded, classified or secret text message to his group member. Free and open source Stego tools are easily available on the internet. Hackers mostly use these free tools randomly to send encoded messages. A list of most common free and open source Stego tools are given at Appendix-I. It is almost impossible to detect a Stego image visually. Forensic investigation is always a time constraints job. Timely and sensible decoded information gives the lead to apprehend the actual culprits. So most of the cases the investigator misses these Stego images due to unavailable of inclusive Stego detection tool. Another limitation to detect Stego images is that mostly Stego images are decoded and detect with the same tool with which it is encoded. A need of sophisticated Stego detect tool that can identify the encoded images and recover the secret messages embedded with any common free Stego tool was mandatory. SSDetect uses some commonly used Stego algorithms such as (LSB encoding, DCT encoding and DWT encoding) and it has also been incorporated several free open source Stego codes. SSDetect also has the ability to incorporate any other Stego algorithm in the utility. All the common free Stego tools do not have the ability to scan the complete drive or folder to identify the Stego image. SSDetect can also scan complete logical drive of a computer or a specific folder to scan for Stego images.

For testing purpose a short secret message is embedded in 200 images of different file images formats (*. Jpg, *. png, *. Tiff) are saved at different location in a computer logical drive of capacity (120 GB, 80 GB and 40 GB) and folder of size (300 MB, 100 MB and 50 MB). Different Stego algorithm with different Steganography tools

(StegDetect, VSL, F-5) is used for testing purpose. The scan results with the detection ration and time taken for the scan are listed in below table.

Table 3 Scan result for Stego detection

Drive folder or	Drive capacity or folder size	Number of files scanned	Number of Stego files	True Positive	True Negative	False Positive	Time taken to complete scan	Detection ration
Logical drive	120 GB	11250	27	25	2	0	13 min 25 sec	93 %
Logical drive	80 GB	8254	19	18	1	5	9 min 48 sec	95 %
Logical drive	40 GB	1570	13	13	0	0	7 min 02 sec	100 %
Folder	300 MB	758	9	8	1	2	5 min 13 sec	89 %
Folder	100 MB	347	7	7	0	0	4 min 04 sec	100 %
Folder	50 MB	98	3	3	0	1	1 min 13 sec	100 %

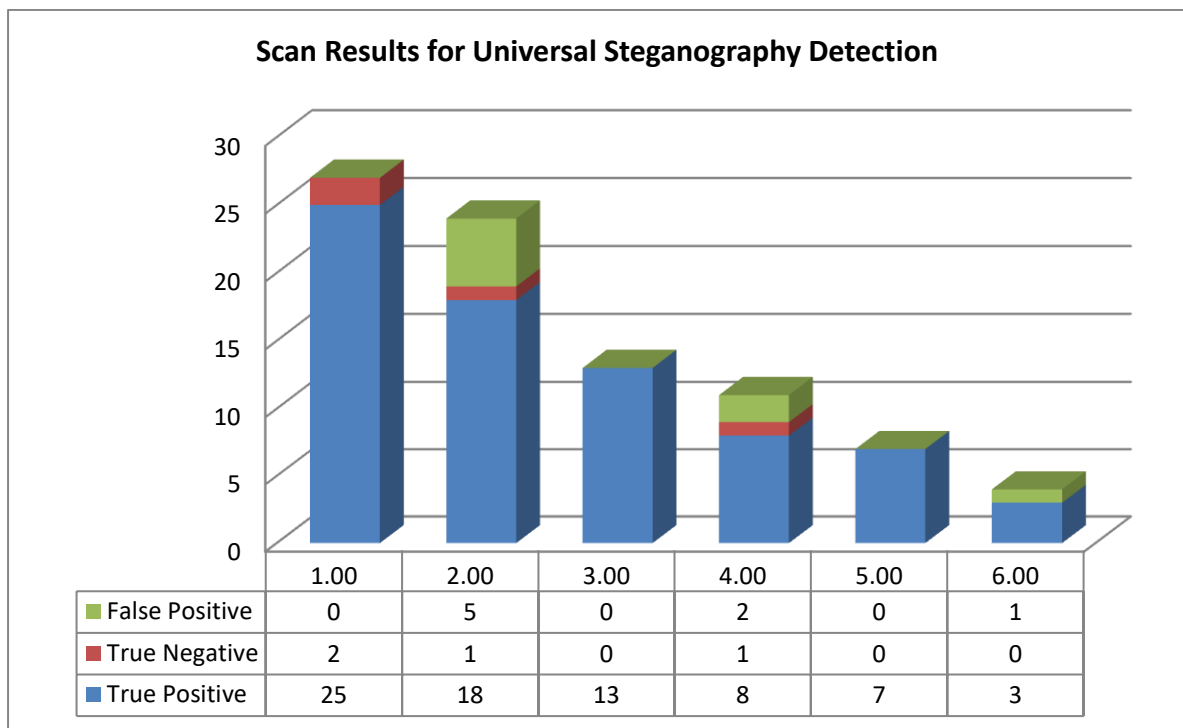


Figure 4-3 Scan Results for Universal Steganography Detection

It has been observed experimentally that detection of developed software is between 80 to 90 percent for all type of image format and all type of image algorithms. In few cases, the detection fails due to very small image size or very robust Stego algorithm. At the end of the scanning, SSDetect will give a scan result which will include the numbers of files scan and time taken to scan the drive or folder. This can result can be saved or export as a text file for further reference.

4.6 SSDetect Comparison with other tools

Performance of SSDetect software could also be evaluated by comparing this tool with a few well-known software available on the internet. Although there is no generic software available that performs automatic detection and decoding of an algorithm. SSDetect is compared with Jphide, F-5, camouflage and Outgues. A brief description of each of these tools is given in Appendix I. The comparison results are given in below tables. Total of 300 images 100 of each file formats are created with the different steganography software and tested with following Steganalysis applications.

Table 4 SSDetect Comparison

Image format	Image size	True-Positive detection				
		Jphide	F5	Camouflage	Outgues	SSDetect
Jpeg	1 kb – 100 kb	10 %	15 %	25 %	17 %	73 %
	101 kb – 200 kb	12 %	17 %	21 %	24 %	81 %
	201 kb – 500 kb	15 %	21 %	18 %	19 %	82 %
	501 kb – 1 mb	15 %	22 %	23 %	21 %	79 %
Png	1 kb – 100 kb	21 %	22 %	21 %	22 %	79 %
	101 kb – 200 kb	19 %	21 %	19 %	21 %	81 %
	201 kb – 500 kb	22 %	23 %	17 %	19 %	82 %
	501 kb – 1 mb	25 %	23 %	20 %	24 %	73 %
Tiff	1 kb – 100 kb	15 %	21 %	18 %	21 %	91 %
	101 kb – 200 kb	15 %	22 %	23 %	19 %	90 %
	201 kb – 500 kb	19 %	21 %	19 %	24 %	92 %
	501 kb – 1 mb	22 %	23 %	17 %	19 %	90 %

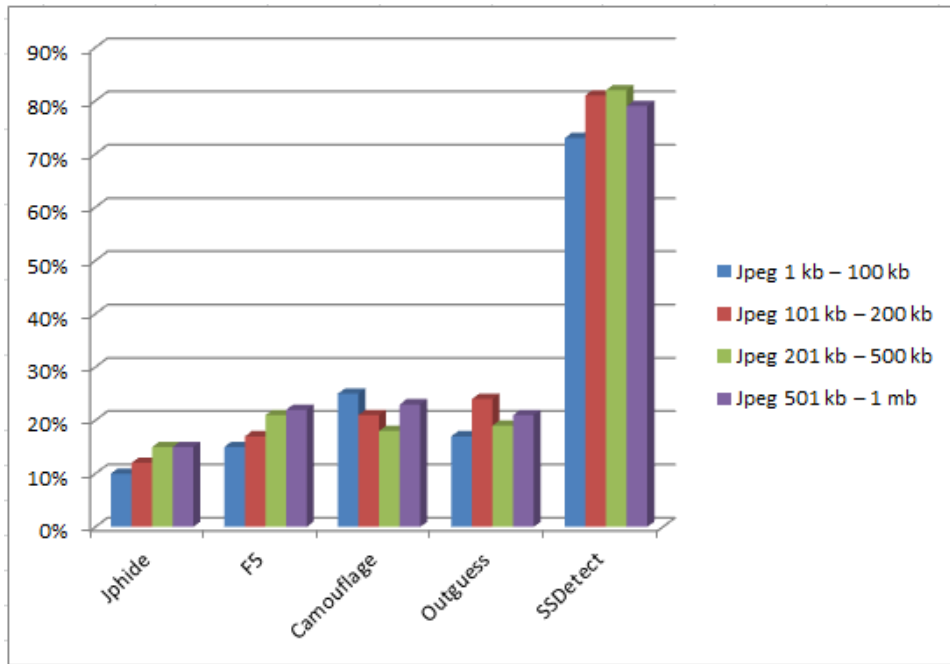


Figure 4-4 Performance graph of SSDetect for JPEG images

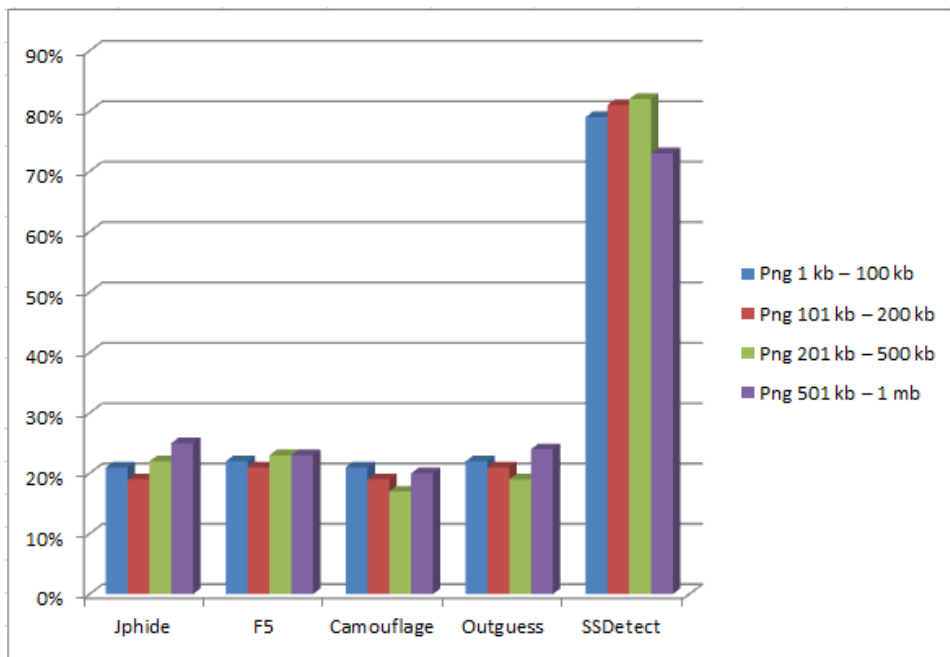


Figure 4-5 Performance graph of SSDetect for PNG images

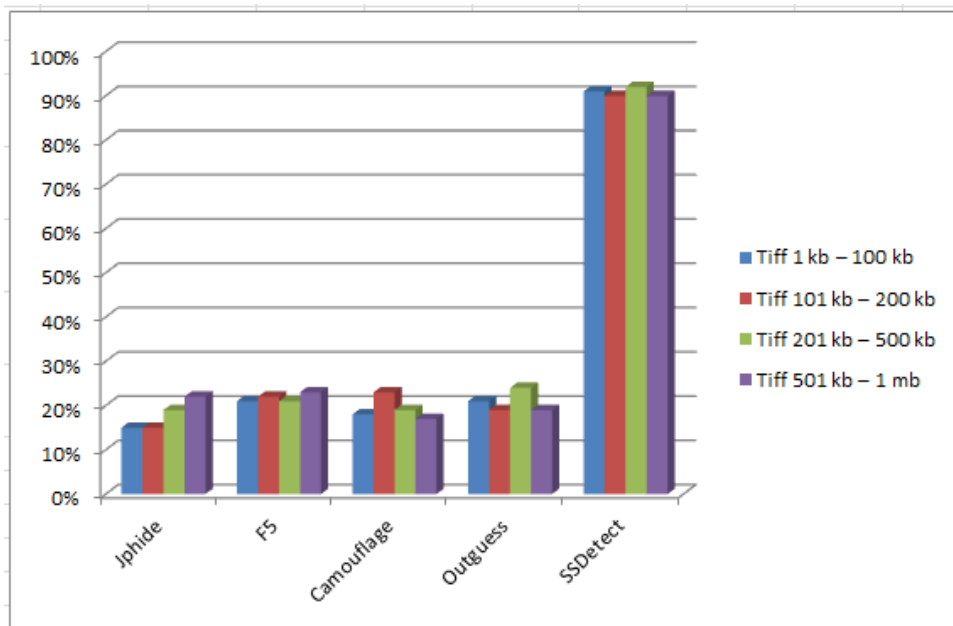


Figure 4-6 Performance graph of SSDetect for TIFF images

4.7 Detect altered file extension and Stegno images at same time

Sometimes the suspect embeds steganography and file extension alteration at the same time. In this case the investigator has to recover the actual file extension and recover the Stegno message at the same time. SSDetect scan the complete drive for the altered file extension and detect steganography at the same time.

4.8 Image encoding – Steganography

In order to verify the detection of SSDetect image encoding or steganography module is also embedded in the developed tool. Image encoding gives the option to the user to select steganography encoding algorithm i.e. LSB, DCT and DWT. The secret message can be export from any text file or the user can write a short message in the provided text box. When the user selects it to cover image it will be displayed in the image box with some basic information i.e. image dimension and image size. After encoding the secret message, the coded image will also be displayed to verify the perceptibility of the image. If the perceptibility of the image is degraded the user can select another image as a cover image or decrease the size of the secret message. All the results used in this thesis have been generated from SSDetect encoding utility.

Screenshot of the encoding window of SSDetect with available options is shown below.

4.9 Working of hybrid stego detect model

The developed program is verified for different scenarios and a few live screenshots of the program are added in this thesis.

Figure 4-7 at page 445 is new case window of the application, each time when user starts the program the user has to fill some mandatory fields mentioned with an asterisk. If the user intentionally or by mistake leaves the mandatory fields empty, the user will not proceed for next modules window. For user convenience investigator name and company name will be filled with previous history names, the user can select these names from dropped down history if the information is same. In the comments field the user can add some remarks or notes regarding the case.

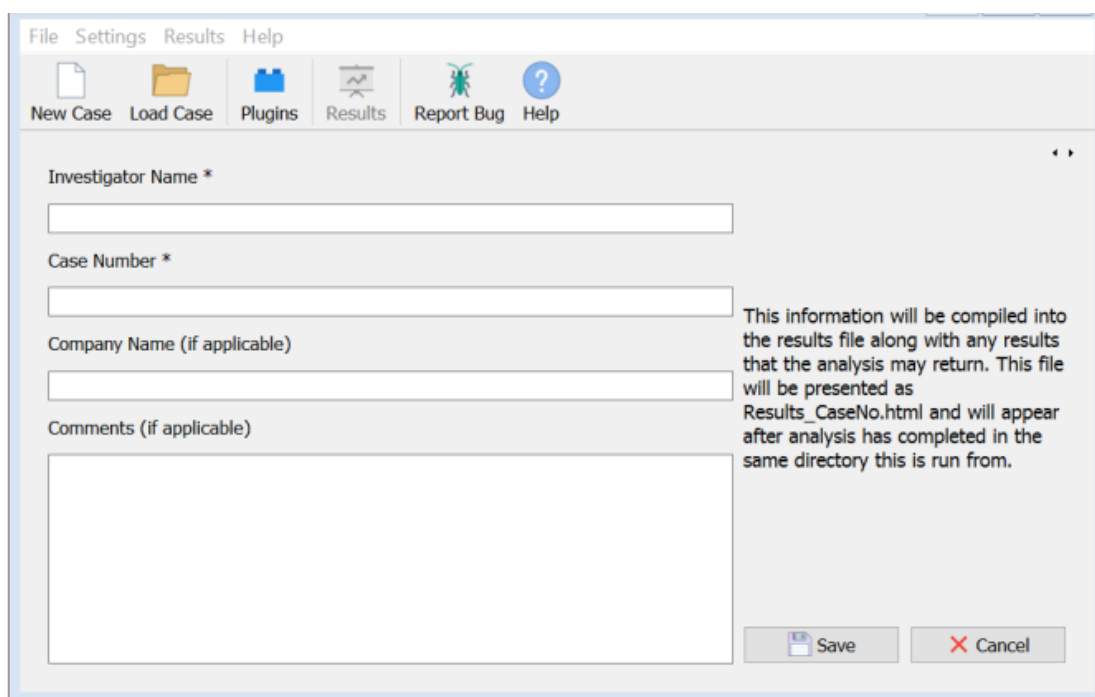


Figure 4-7 Screenshot of New Case window

After completing the case information the next window is the analysis screen as shown in **Figure 4-8**. User can select a single file or complete directory for scanning stego images. After selecting one of the options single file or complete directory the user will click start analysis button. After clicking on start analysis button a process

bar will be appearing at the left end of the screen. It is very obvious that the analysis time is proportional to the numbers of the files in the directory.

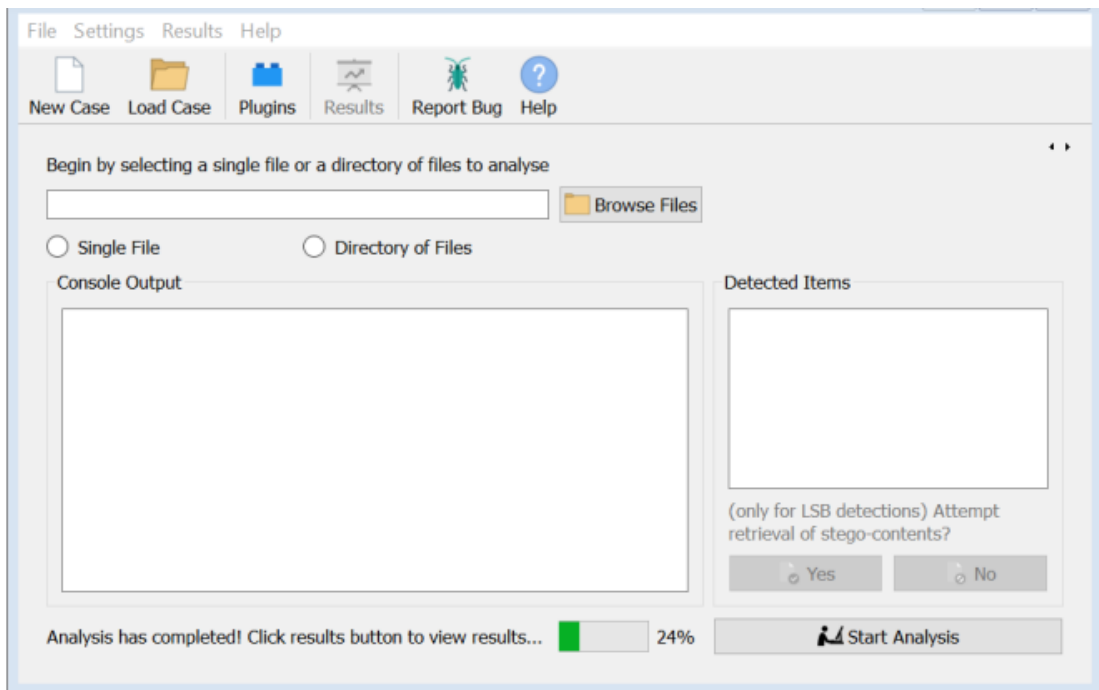


Figure 4-8 Analysis Window of SSDetect

The below screen shows the list of images files in the mentioned directory.

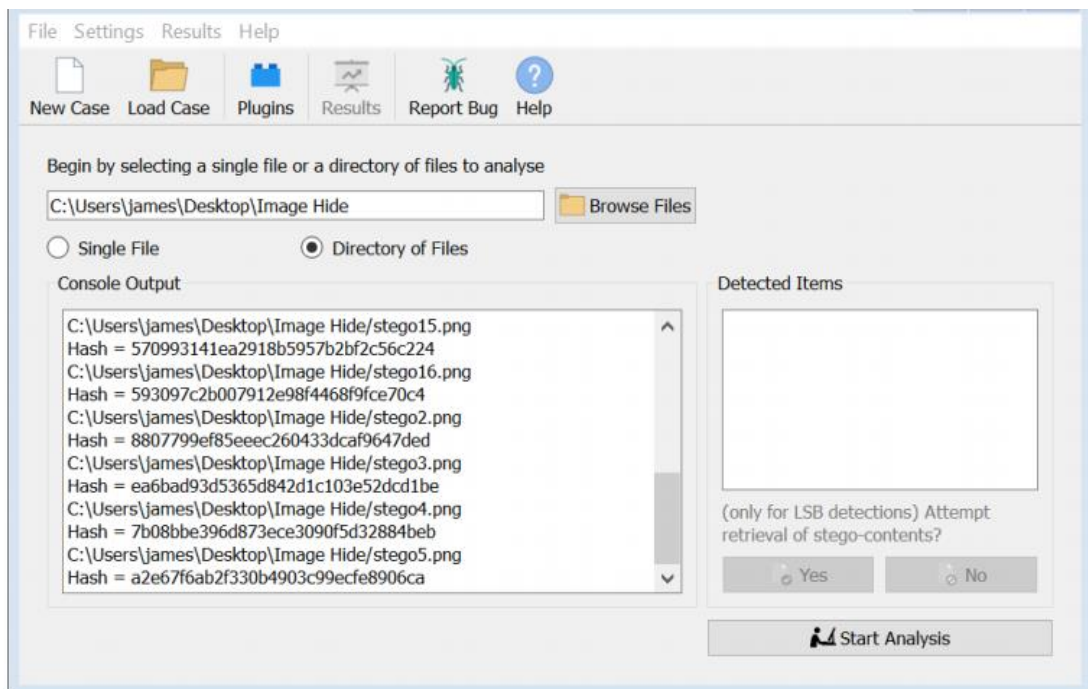


Figure 4-9 Analysis Process Window

The below two screens show the analysis process. The left field in the immediate below image shows the loaded images while at the right field shows the list of true positive images for which the stego has been detected. Once the analysis process has been completed the show result button will then be active.

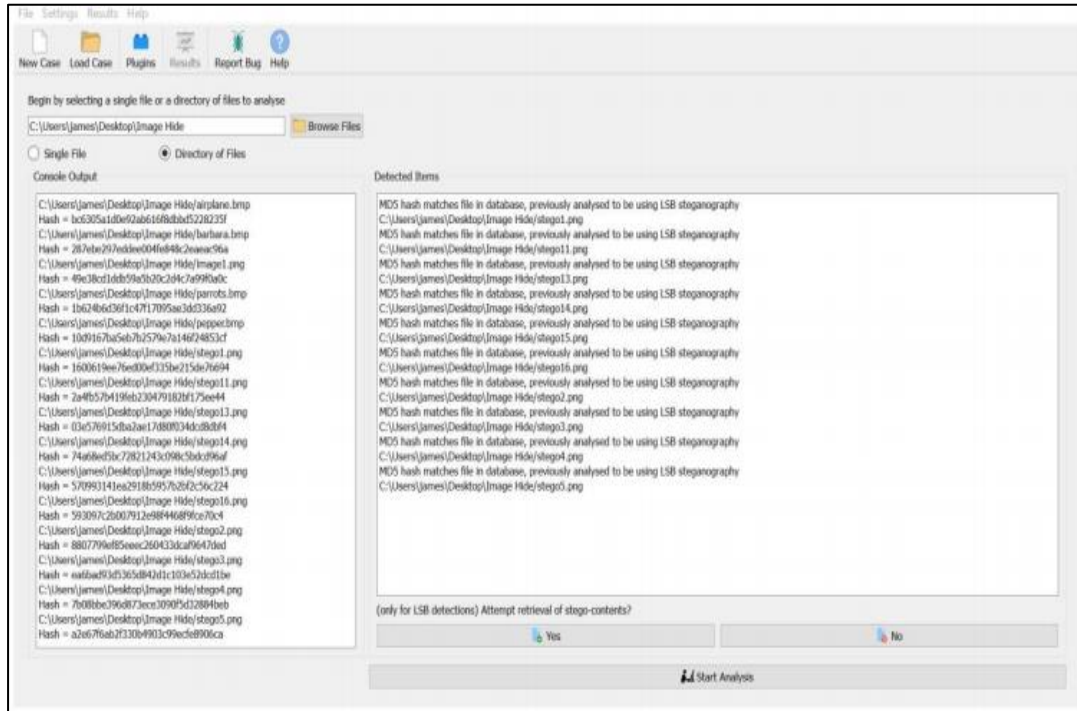


Figure 4-10 SSDetect Analysis Window - 1

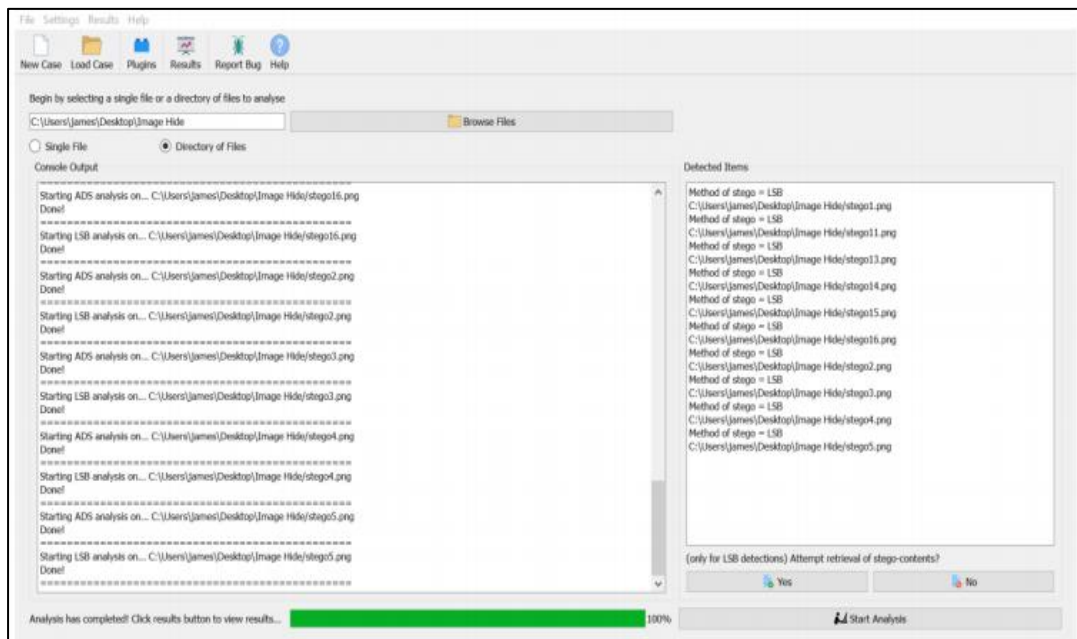


Figure 4-11 SSDetect Analysis Window – 2

4.10 Conclusion

After successful development of algorithm in the line of proposed elucidation, the next phase is the evaluation. Large no of test data has been inhabited to create a practical scenario. While creating a testing data set practical challenges kept in mind. How the developed algorithm is testified on given data set are explained in this chapter. All the results and grades are plotted in tables / charts to testify the objective of thesis.

CONCLUSION AND FUTURE DIRECTION

5.1 Introduction

Image steganography has become favorite method among terrorists and bad guys to exchange secret information stealthily. With the passage of time techniques for Steganography become appraised and diversified. Every author of code tries to write a robust and imperceptible Steganography algorithm. This diversity in algorithm creates a gap between Steganography and Steganalysis programs. So the immense variety of data hiding techniques create a problem for a digital investigator in Forensic Lab to identify a few images among thousands of images in which the coded text is available and properly decode the hidden message. Several available Steganalysis attacks can be broadly divided into two groups, Signature attack and Statistical attack. Unfortunately, there is no inclusive Steganalysis software available that can detect and decode many Stenographic techniques.

5.2 Research flow

The contribution and research this complete thesis is divided into three segments. In the first section a complete research is conducted for magical bytes (file headers) that are used to identify a file and its responding application. A comprehensive list of common files headers and their position in file are created. An efficient code to compare the list of file header with the given files (file without extensions) is developed. This module of code is testified for several of hundreds of files and gives results. This research saves the time of Forensic Investigator, who has to identify the actual extension of file manually by decoding the file in hex viewer application.

The second and third deliverables of this research thesis are to develop an image steganography and Steganalysis application with generic features explained in chapter 1. Steganalysis was the prime task of the research but the steganography module was developed to create a test dataset for the evacuation of decoding module. When I select this topic for my thesis, I thought it will be a simple project. I had to study a few common stenographic algorithms, most of these algorithms will be same and I needed to integrate these algorithms to develop steganography reverse software. As I started the research I found different algorithms and most of the authors do not disclose their

codes. There are many stego reverse attacks, some of Steganalysis attack are very complicated and lengthy algorithms. After thorough research, I shortlisted some very common and mostly used algorithms (StegDetect, F5 and VSL). Very intensive literature was studied for selected stego reverse attacks which I have added some attacks that can decode most of the steganography algorithms.

The developed program is then evaluated and compared with state of the art and well-reputed software. The results of this developed tool are found better and satisfactory. The main deliverable of this thesis was to create a universal, hybrid, efficient, user-friendly program. The comparative analysis of developed program with different well-reputed software is given in chapter 4 of this thesis.

5.3 Research limitation

In spite of having a valuable contribution in the knowledge of image Steganography and Steganalysis, there is always some limitation and available space for future improvement. One limitation in the developed SSDetect software is, its recovery from high distorted or noise polluted images is not very good.

As several decoding algorithms run for one image, so it is a time taking the task. To get efficient results in the limited time it is recommended to scan single file or few files at a single time, instead of scanning a whole directory.

In some cases, a password key is used to encode a secret message, this application has no capability to recover the encoded key and encoded the message.

5.4 Future Direction

There is always a space of improvement available in any research. This research work could be improved if it can recover hidden messages from password encoded files. Block diagram of image steganography with security key used for encoding and decoding are shown in figure 5.1. Password are used to implement another layer of security to the embedded secret text.

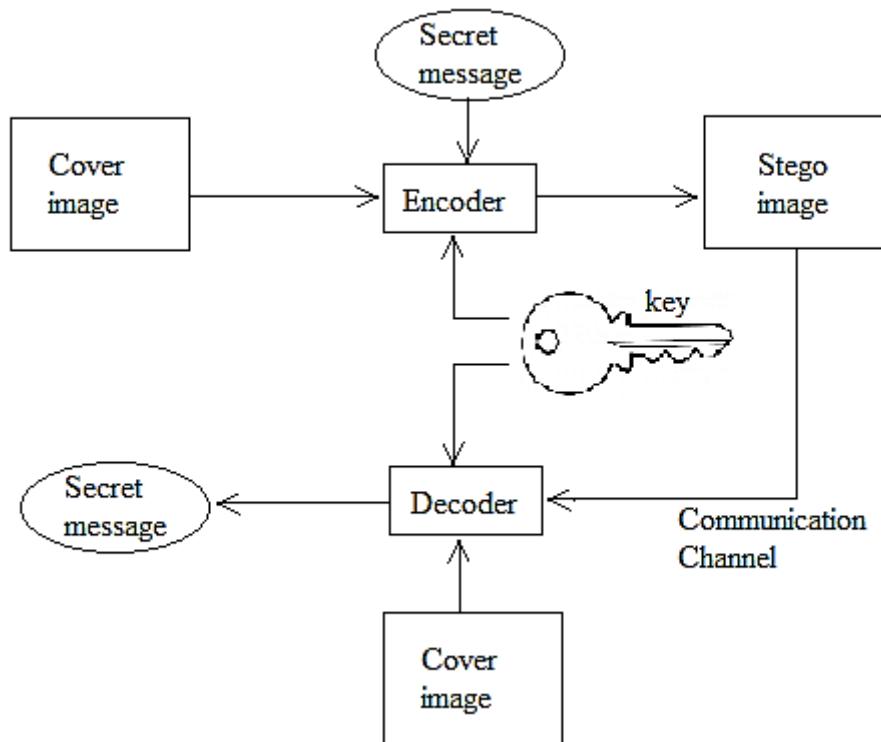


Figure 5-1 Password Protected Image Steganography

To identify the methods that how these passwords can be recover is another valuable contribution.

Another area of improvement in this thesis is if we can enhance its recovery from distorted or noisy images. When a stego image is transported over a noisy and non-secure channel, several clutter contaminate the image. When an image is sent over computer / mobile application (Emails, WhatsApp etc.) its size could be downgrade. Downgrading in size of an image discard many information in image. Examples of several degradations / distortion in images are listed in figure 5.2.

Sometimes image is intentionally distorted or contaminated with some noise to make difficult the recovery. So an intensive research is required to decode the hidden message from noisy images.

This research was to recover the hidden message from image format only, however many other medium i.e. audio, video and text are also as cover medium. Recovery of embedded text from other medium could also be incorporated in this research.

One module of the research is to recover the extensions of file formats when these extensions are altered or removed. In the developed algorithm the flexibility to add file signature are merged. A rich database of all common file formats should be maintained to get the improved result.



Figure 5-2 Examples of Degradation or Distortion in image

It is difficult to acquire the embedded message from a robust steganography algorithm. Every new algorithm is more robust and secure that have the capability to embed large payload, so a continuous research to develop a Steganalysis algorithm that recover the coded message from a highly robust algorithm are required.

Signature-based image Steganography algorithms are also improving; every new algorithm tries to conceal the signature from the analysis. In the current algorithm it is required to add new techniques to recover the signature.

5.5 Conclusion

This chapter start with the introduction of image steganography, its importance in present cyber environment for LEA and government organization. Required area of research is elaborated in the domain of digital forensic investigators. Several limitations are highlighted in the present solution. In this chapter we also present complete workflow for research to achieve the proposed solution. Research limitations due to time constraints or technology constraints are also discussed in this chapter. Future direction to enhance the capability of developed algorithm are also expounded.

APPENDICES

Appendix I

Directory of steganography software							
Application name	Availability (Freeware / preparatory)	Source code available (Yes / No)	Image format supported	Interface type (GUI / Command Line)	Platform (Windows / Linux)	URL / link	Remarks
Blindside	Freeware.		BMP	Command line	Linux	http://www.blindside.co.uk	
BMP Secrets	Freeware.			GUI	windows	http://www.pworlds.com/products/i_secrets.html	High capacity
Camouflage	Freeware	Yes		GUI	windows	(http://www.camouflagesoftware.co.uk/)	EOF insertion
Contraband	Freeware	Yes	BMP	GUI	Windows	http://www.biol.rug.nl/hens/j/contrabd.exe	
DPT (Data Privacy Tools)	Freeware	No	BMP	GUI	Windows	http://www.xs4all.nl/~bernard/home_e.html	
Encrypt pic	\$10		BMP	GUI	windows	ftp://ftp.elet.polimi.it/mirror/Winsite/win95/miscutil/encpic13.exe	
EZStego	Freeware	Yes	GIF	GUI	Linux	http://www.stego.com	
FFEncode	Freeware	No		Command line	Windows	http://www.rugley.demon.co.uk/security/ffencode.zip	
Hide4PGP	Freeware	Yes	BMP, WAV, VOC	Command line	Windows / Linux	http://www.heinz-repp.onlinehome.de/Hide4PGP.htm	
Hide and Seek	Freeware		GIF	Command line	Windows	http://www.rugley.demon.co.uk/security/hds_k50.zip	
Invisible Secrets	Freeware		JPG, BMP, PNG	Browser	Windows	http://www.innova-tools.com/software/isecrets	
JSteg	Freeware	yes	JPG	Command line	windows	http://linkbeat.com/files	
OUTguess	Freeware	Yes	JPG, PNG	Command line	Windows	http://www.outguess.org	
Stegdetect	Freeware	Yes	JPEG	GUI	Windows / Linux	http://www.outguess.org/detec	Automated

						tion.php	
Steghide	Freeware	Yes	BMP, WAV and AU	Command line	Windows / Linux	http://www.croswinds.net/~shetzl/steghide/index.html	
S-Tools	Freeware	Yes	BMP, GIF, WAV		Windows	ftp://ftp.ntua.gr/pub/crypt/mirrors/idea.sec.dsi.unimi.it/code/s-tools4.zip	
TextHide	Commercial ware \$39	No		GUI	Windows	http://www.texthide.com)	

Appendix II

TECHNICAL COMPARISONS OF IMAGE FILES

Parameters	.JPEG	.PNG	.TIF
Compression algorithm	Lossy (and partly lossless), DCT, RLE, and Huffman predictive nearest neighbor	Lossless and DEFLATE	None, LZW, RLE, ZIP, and other
Raster vector	Raster	Raster	Both
Color depth	8-bit (greyscale), 12-bit, and 24-bit	bitmap (1bpc), indexed (8bpc), grayscale and RGB (8bpc, 16bpc)	1,2,4,8,16,24 and 32
Indexed color	No	Yes	Yes (1-8 Bit Modes)
Transparency	No	Yes; indexed, grayscale and RGB	Yes
Metadata	Yes	Yes	Yes
Interlacing	Yes	Yes, Adam7 Algorithm	Yes, for JPEG compression
Multipage	No	No	Yes
Animation	No	No	No
Layers	No	No	Yes
Color Management	Yes	Yes	Yes
Extendable	No	Yes Via Chunks	Yes, via tags
HDR Format	Unofficial	No	Yes, TIFF float

GENERAL COMPARISON OF COMMON IMAGE FILES FORMAT

Parameters	.JPEG	.PNG	.TIF
Full Name	Joint Photographic Experts Group	Portable Network Graphics	Tagged Image File Format
Owner	Joint Photographic Experts Group	World Wide Web Consortium	Adobe Systems
File Extension	.jpg, .jpeg, .jpe (containers: .jif, .jfif, .jfi)	.png	tiff, .tif
MIME Type	Image/ jpeg	Image / png	image/tiff
Application	Photographic images. Supported by most web browsers.	W3C endorsed a replacement for GIF. Supported by most web browsers.	Document scanning and imaging format, also functions as a container.
Patented	Expired	No	No

Appendix III

SIGNATURE / MAGIC BYTES COMMON FILES

Ser	File Type	Details	Hex Signature
1.	Avi	Audio video interleave	52 49 46 46 41 56 49 20
2.	Bmp	Bitmap	42 4D
3.	Doc	Compound File Binary Format, a container format used for document by older versions of Microsoft Office	D0 CF 11 E0 A1 B1 1A E1
4.	Docx	Other file formats based on zip file format	50 4B 03 04 Or 50 4B 05 06 Or 50 4B 07 08
5.	Exe dll	DOS MZ executable	4D 5A
6.	Gif	Image file encoded in the graphics interchange format	47 49 46 38 37 61
7.	Ico	Computer icon	00 00 01 00
8.	Jpg	JPEG raw or in the JFIF or Exif file format	FF D8 FF DB
9.	Mp3	MP3 file with an ID3v2 container	49 44 33
10.	Pdf	Pdf document	25 50 44 46
11.	Png	Image encoded in the Portable Network Graphics format	89 50 4E 47 0D 0A 1A 0A
12.	Rar	RAR archive version 1.50 onwards	52 61 72 21 1A 07 00
		RAR archive version 5.0 onwards	52 61 72 21 1A 07 01 00
13.	Swf	flash .swf	43 57 53
14.	Tiff	Tagged Image File Format	49 49 2A 00 (little endian format) 4D 4D 00 2A (big endian format)
15.	Wav	Waveform Audio File Format	52 49 46 46 57 41 56 45
16.	3gp	3rd Generation Partnership Project multimedia files	66 74 79 70 33 67

BIBLIOGRAPHY

- [1] Neil F and Jajodia, Sushil Johnson, "Exploring Steganography: Seeing the unseen," *Computer*, 1998.
- [2] Walter and Butera, William and Gruhl, Daniel and Hwang, Raymond and Paiz, Fernando J and Pogreb, Sofya Bender, "Applications for data hiding," *IBM systems journal*, 2000.
- [3] Hamza and Sankur, B, "Detection of audio covert channels using statistical footprints of hidden messages," *Digital Signal Processing*, 2006.
- [4] David Frith, "Steganography approaches, options, and implications," *Network Security*, 2007.
- [5] Hany Farid, "Image forgery detection," *IEEE Signal processing magazine*, 2009.
- [6] Jessica Code, "Unseen: An Overview of Steganography and Presentation of Associated Java Application C-Hide," *Citeseer*, Retrieved January, 2009.
- [7] Byungho and Kim, Dukyun and Shin, Daecheol Park, "A Study on a Method Protecting a Secure Network against a Hidden Malicious Code in the Image," *Indian Journal of Science and Technology*, 2015.
- [8] Guillermo, Juan E. Tapiador, and Pedro Peris-Lopez Suarez-Tangil, "Stegomalware: Playing hide and seek with malicious components in smartphone apps," *International Conference on Information Security and Cryptology*. Springer, Cham, pp. 496-515, 2014.
- [9] Conway Maura, "Code wars: Steganography, signals intelligence, and terrorism Knowledge," *Technology & Policy* 16.2, pp. 45-62, 2003.
- [10] Andrew D and Bas, Patrick and B, "Moving steganography and steganalysis from the laboratory into the real world," *Proceedings of the first ACM workshop on*

Information hiding and multimedia security, 2013.

- [11] Andreas Westfeld, "F5—A Steganographic Algorithm," *International workshop on information hiding*, 2001.
- [12] Clarke D, *Technology and Terrorism.:* Transaction Publishers, 2004.
- [13] Siwei and Farid, Hany Lyu, "Steganalysis using higher-order image statistics," *IEEE Transactions on Information Forensics and Security*, 2006.
- [14] Simon Singh, *The code book: the science of secrecy from ancient Egypt to quantum cryptography.:* Anchor, 2000.
- [15] (2017, Mar.) Operating system market share. [Online]. [\url](#)
- [16] [Online]. nationalarchives.gov.uk/pronom/default.aspx
- [17] [Online]. nationalarchives.gov.uk/information-management/our-services/dc-file-profiling-tool.htm
- [18] Harjit Singh, "Analysis of Different Types of Steganography," 2016.
- [19] Mennatallah M., Amal S. Khalifa, and Mostafa GM Mostafa Sadek, "Video steganography: a comprehensive review," *Multimedia tools and applications* 74.17 , pp. 7063-7094, 2015.
- [20] Hemant, and Setu Chaturvedi Gupta, "video steganography through LSB based hybrid approach," *International Journal of Computer Science and Network Security (IJCSNS)* 14.3, 2014.
- [21] Patrick and Fridrich, Jessica Pevny, "Steganalysis by subtractive pixel adjacency matrix," *IEEE Transactions on Information Forensics and Security*, 2010.
- [22] Abbas and Condell, Joan and Curran, Kevin and Mc Kevitt, Paul Cheddad, "A secure and improved self-embedding algorithm to combat digital document forgery," *Signal Processing*, 2009.

- [23] Shreyank N Gowda, "Dual layered secure algorithm for image steganography," *Applied and Theoretical Computing and Communication Technology (iCATccT), 2nd International Conference on. IEEE*, 2016.
- [24] Shreyank N., and Sumit Sulakhe Gowda, "Block-Based Least Significant Bit Algorithm For Image Steganography," *Annual Int'l Conference on Intelligent Computing, Computer Science & Information Systems (ICCSIS-16)*, 2016.
- [25] Jagdish, Viraj Sonawane, and R. N. Awale Mali, "Image steganography using block level entropy thresholding technique," *Journal for Research/ Volume 2.04*, 2016.
- [26] Vidyasagar M and Chang, Elizabeth Potter, "Grey level modification steganography for secret communication," *Industrial Informatics, 2004. INDIN'04. 2004 2nd IEEE International Conference on*, 2004.
- [27] Khan Muhammad, "Evaluating the Suitability of Color Spaces for Image Steganography and Its Application in Wireless Capsule Endoscopy," *Platform Technology and Service (PlatCon), 2016 International Conference on. IEEE*, 2016.
- [28] Da-Chun and Tsai, Wen-Hsiang Wu, "A steganographic method for images by pixel-value differencing," *Pattern Recognition Letters*, 2003.
- [29] Hemang A., and Dr Nehal G. Chitaliya Prajapati, "Secured and Robust Dual Image Steganography: A Survey," *International Journal of Innovative Research in Computer and Communication Engineering 3.1* , 2015.
- [30] Gandharba and Lenka, Saroj Kumar Swain, "Steganography using two-sided, three-sided, and four-sided side match methods," *CSI transactions on ICT*, 2013.
- [31] Brian and Wornell, Gregory W Chen, "Quantization index modulation: A class of

- provably good methods for digital watermarking and information embedding ," *IEEE Transactions on Information Theory*, 2001.
- [32] Hsien-Chu and Wang, Hao-Cheng and Tsai, Chwei-Shyong and Wang, Chung-Ming Wu, "Reversible image steganographic scheme via predictive coding," *Displays*, 2010.
- [33] Vojtěch, and Jessica Fridrich Holub, "Low-complexity features for JPEG steganalysis using undecimated DCT," *IEEE Transactions on Information Forensics and Security* 10.2, pp. 219-228, 2015.
- [34] Manju, and Dr Harish Rohil Parul, "Optimized image steganography using discrete wavelet transform (DWT)," *International Journal of Recent Development in Engineering and Technology (IJRDET)* 2.2, pp. 75-81, 2014.
- [35] Sumeet, Savina Bansal, and Rakesh K. Bansal Kaur, "Steganography and classification of image steganography techniques," *Computing for Sustainable Global Development (INDIACom), 2014 International Conference on. IEEE*, 2014.
- [36] Hedieh Sajedi, "Steganalysis based on steganography pattern discovery," *Journal of Information Security and Applications* 30, pp. 3-14, 2016.
- [37] Neil F., Stefan Katzenbeisser Johnson, "A survey of steganographic techniques," *Information Hiding*, pp. 43-78, 2000.
- [38] Mahmoud Al Qutayari, Hassan Barada Tariq Al Hawi, "Steganalysis attacks on stego images using stego-signatures and statistical image properties," *TENCON 2004, Region 10 Conference Vol: 2*, pp. 104-107, 2004.
- [39] Jessica, Miroslav Goljan, and Rui Du Fridrich, "Practical steganalysis of digital images-state of the art," *Proc. SPIE Photonics West, Electronic Imaging, Vol*

4675, pp. 1-13, 2002.

- [40] M. Goljan, R. Du J. Fridrich, "Steganalysis based on JPEG compatibility," *SPIE Multimedia System and Applications IV*, pp. 275-280, August 20–24, 2001.
- [41] Yambem Jina, Kh Manglem Singh, and Themrichon Tuithung Chanu, "Image steganography and steganalysis: A survey," *International Journal of Computer Applications* 52.2, 2012.
- [42] Der-Chyuan, and Chen-Hao Hu Lou, "LSB steganographic method based on reversible histogram transformation function for resisting statistical steganalysis," *Information Sciences* 188, pp. 346-358, 2012.
- [43] R. Du, L. Meng J. Fridrich, "Steganalysis of LSB encoding in color images," *IEEE Int. Conf. on Multimedia and Expo, New York*, July 31–August 2, 2000.
- [44] N. Memon, B. Sankur I. Avcibas, "Image steganalysis with binary similarity measures," *IEEE Int. Conf. on Image Processing, Rochester, New York*, September 2002.
- [45] A., Condell, J., Curran, K., & Mc Kevitt Cheddad, "Digital image steganography: Survey and analysis of current methods," *Signal Processing* 90.3, pp. 727-752, 2010.
- [46] Hongxun Yao, Wen Goa Shaohui Liu, "Steganalysis of data hiding techniques in wavelet domain," *IEEE Int. Conf. on Information Technology*, 2004.
- [47] Resul Das, "An Investigation on Information Hiding Tools for Steganography," *International Journal of Information Security Science* 3.3 , pp. 200-208, 2014.
- [48] Omed S., Julio C. Hernandez-Castro, and Benjamin Aziz Khalid, "A study on the false positive rate of Stegdetect," *Digital Investigation* 9.3, pp. 235-245, 2013.
- [49] Andreas Westfeld, "F5—a steganographic algorithm," *Information hiding*.

Springer Berlin/Heidelberg, pp. 289-302, 2001.

- [50] A. D., Bas, P., Böhme, R., Cogramne, R., Craver, S., Filler Ker, "Moving steganography and steganalysis from the laboratory into the real world," *ACM workshop on Information hiding and multimedia security, Montpellier, France*, 2013.
- [51] J., Goljan Fridrich, "practical steganalysis of digital images: state of the art ," *SPIE Photonics West, Electronic Imaging, CA*, 2002.
- [52] J.P., Pollitt, M., & Swauger, J. Craiger, *Law enforcement and digital evidence, Handbook of Information Security*. New York, USA, 2005.