Synopsis of Project work on,

"Efficient data hiding system using LZW, Cryptography and Image Steganography with a hybrid approach"

Submitted in partial fulfillment of the requirements

of the degree of

(Bachelor of Engineering)

by

Amit S. Singh

Nayan Solanki

Supervisor:

Prof. Foram Shah

(Computer Engineering)

AET's

Atharva College of Engineering

(2017-18)

AET'S

ATHARVA COLLEGE OF ENGINEERING

CERTIFICATE

This is to certify that

Amit S. Singh

Nayan Solanki

Have satisfactorily completed the requirements of the Synopsis

On

"Efficient data hiding system using LZW, Cryptography and Image Steganography with a hybrid approach"

As prescribed by the *University of Mumbai for academic year* ***2017-18***

Under the guidance of

Prof. Foram Shah

Prof Deepali Maste	**Prof Mahendra Patil**	**Dr S.P.Kallurkar**
Project Coordinator	**Head(Computer Engineering)**	**Principal**

Internal Examiner	**College Seal**	**External Examiner**

Index

List of Figures

Abstract

Secure data transmission over internet, it is important to transfer data in high security and high confidentiality, information security is the most important issue of data communication in networks and internet. To secure transferred information from intruders, it is important to convert information into cryptic format. Different methods used to ensure data security and confidentiality during transmission like steganography and cryptography.

This project improves information security through developing efficient compression of texts and image cryptography algorithm by using encryption with steganography. The proposed algorithm ensures the encryption and decryption using AES (Advanced Encryption Standards) as well as compression and decompression by using LZW and RGB pixel shuffling with steganography by using least significant Bit (LSB) method that make way to insert data bits in LSB bits of RGB pixels of cover image. Hybrid Approach can reduce the time of transmission of data. These algorithms are performed by using JAVA program.

1.Introduction

1.1 Need

- Current steganography project is having complexity problem in terms of time per Data. Now Days Steganography project can hide data in equal length of image file, For Example It can hide 1Mb data with At least (Or More than) 1Mb Image file, it means that For Extracting 1Mb data we have to download At least (OR more than) 2Mb file (i.e. 50% loss of Data OR more than 50% loss).
- And also, its cost of transmission is more. Second problem is, it takes more time for a complete transmission from sender to receiver.

1.2 Basic Concept

- LZW compression scheme is used to optimize the size of secret data, it will enable a person to hide approx. 2 times more data in a cover-image, i.e. Now we can store Double Amount of Data into Same Image File. This Way we can reduce the cost of data and we can also reduce loss of data up to 50%. This approach is secure against the RS detection attack and its steno-image is totally indistinguishable from the original image (cover-image) by the human eye.
- And By using Hybrid Approach we can reduce transmission time by Extracting data parallelly at decoding time. This way we can reduce some amount of transmission time.

1.3 Applications

Applications of this system in following areas: -

- Confidential communication and secret data storing
- Protection of data alteration
- Access control system for digital content distribution
- Media Database systems

2. Review of Literature

Paper 1:

Efficient Data Hiding System using LZW, Cryptography and GIF Image Steganography

Intisar Majeed Saleh, Hanna Hameed Merah
AL. Rubidian University College
Baghdad, Iraq.

Content referred:

The combination of steganography and cryptography is considered as one of the best security methods used for message protection, due to this reason, in this paper, a data hiding system that is based on image steganography and cryptography is proposed to secure data transfer between the source and destination. Animated GIF image is chosen as a carrier file format for the steganography due to a wide use in web pages and a LSB (Least Significant Bits) algorithm is employed to hide the message inside the colors of the pixels of an animated GIF image frames. To increase the security of hiding, each frame of GIF image is converted to 256 color BMP image and the palette of them is sorted and reassign each pixel to its new index, furthermore, the message is encrypted by LZW (Lempel _ Ziv_Welch) compression algorithm before being hidden in the image frames. The proposed system was evaluated for effectiveness and the result shows that, the encryption and decryption methods used for developing the system make the security of the proposed system more efficient in securing data from unauthorized users. The system is therefore, recommended to be used by the Internet users for establishing a more secure communication.

Paper 2:

Combined Strength of Steganography and Cryptography

Aishwarya Baby
P.G Scholar
FISAT,Mookkannooor,Kerala,India

Hema Krishnan
Assistant Professor
FISAT,Mookkannooor,Kerala,India

Content referred:

The use of internet for communication purpose has rapidly increased and it magnified the attacks to users. Protecting the data is a big challenge for computer users. Cryptography and Steganography are widely used techniques to ensure security. Both techniques have many applications in computer science and other related fields. Both methods provide security in their own ways, but to add multiple layers of security it is always a good practice to use combination of these techniques. The concepts of steganography, cryptography and their applications in

the security of digital data communication across network is studied in this paper and technical survey of recent methods which combined steganography and cryptography is presented.

Paper 3:

An Efficient Cryptography Using HASH-LSB Steganography with RC4 and Pixel Shuffling Encryption Algorithms

Assit.Lec. May H.Abood
Computer Engineering dept
College of Engineering, Al-iraqia University
Baghdad, Iraq.

Content referred:

This paper improves information security through developing efficient image cryptography algorithm by using encryption with steganography. The proposed algorithm ensures the encryption and decryption using RC4 stream cipher and RGB pixel shuffling with steganography by using hash-least significant Bit (HLSB) that make use of hash function to developed significant way to insert data bits in LSB bits of RGB pixels of cover image. The security evaluations are presented by calculating a peak signal to noise ratio and mean square error. For secret image, PSNR is infinity and MSE is 0. For cover image, PSNR is about 63 dB and MSE is about 0.03. The results show that high level of the similarity exists between the stage-images and cover images and the same is for secret images and extracted image as represented also in In Histogram Analysis of secret images. These algorithms are performed by using MATLAB program.

3.Existing System

They proposed new approach wherein both cryptography and steganography are used to encrypt the data and also to conceal these encrypted contents in some other medium. In this method, we can secure Message File by converting it into an encrypted text using S-DES algorithm (Or Triple DES, AES, RSA, etc.) and a secret key and then concealing this encrypted text in some other image. This technique has been tested and it has been observed that they prevent the possibilities of stainless also. Now days project of combination of steganography and cryptography either with LZW compression method or with single hybrid approach techniques. In case one they used combine strength of steganography and cryptography with alphabets repetition to compress the file because of this we can save the data to transfer a file over the internet . Average compression of message file is 50% (i.e. just half) And in case two, this also contain the combination of steganography and cryptography but only difference is instead of LZW compression algorithm, it use hybrid approach techniques. Its mean during the extraction of message at receiver end it divides virtually to the whole stego-image file as in two part to decode a message file simultaneously from stego-image file. It saves time of decoding up to 50% of total time of decoding.

4. Aim and Objective

Aim: -

The combine the strength of steganography and cryptography for a secure data and data hiding system that is based on image steganography and cryptography is proposed to secure data transfer between the source and destination. The main focus of this project is to combine the all different trends which are already been proposed in the direction of cryptography and steganography to reduce the time and space in terms of transmission over the network.

Objective: -

- To study the security mechanism in currently existing system.
- To study and develop the module required for steganography to make the

  appearance of secret data invisible to any outside intruder.
- To study and develop the algorithms that can be used for easy encryption and decryption with image steganography.
- To study and develop the modules that can be used to reduce the time and space complexity during transmission over the network.
- To study and develop the modules that can used for lossless compression and decompression method.

5.Problem Statement

Current steganography project is having complexity problem in terms of time per Data. Now Days Steganography project can hide data in equal length of image file, For Example It can hide 1Mb data with At least (Or More than) 1Mb Image file, it means that For Extracting 1Mb data we have to download At least (OR more than) 2Mb file (i.e. 50% loss of Data OR more than 50% loss). And also, its cost of transmission is more. Second problem is, it takes more time for a complete transmission from sender to receiver. Many different methods are also proposed to nullify the above problem. But the solution of that is not completed. It means they haven't succeeded 100%. Use of LZW algorithm to reduce the file size, currently we are using compression in alphabets only, since LZW compression method works on repetition of words, its mean more the repletion less the size of compressed data file. But still as compare to repletion of words in ASCII value in more than repetition in alphabets. And also hybrid approach use separately.

6.Proposed System for Project

The combination of steganography and cryptography is considered as one of the best security methods used for message protection, due to this reason, in this paper, a data hiding system that is based on image steganography and cryptography is proposed to secure data transfer between the source and destination. Animated GIF image is chosen as a carrier file format for the steganography due to a wide use in web pages and a LSB (Least Significant Bits) algorithm is employed to hide the message inside the colors of the pixels of an animated GIF image frames. To increase the security of hiding, each frame of GIF image is converted to 256 color BMP image and the palette of them is sorted and reassign each pixel to its new index, furthermore, the message is encrypted by LZW (Lempel _ Ziv_Welch) compression algorithm before being hidden in the image frames. The proposed system was evaluated for effectiveness and the result shows that, the encryption and decryption methods used for developing the system make the security of the proposed system more efficient in securing data from unauthorized users. The system is therefore, recommended to be used by the Internet users for establishing a more secure communication.

System which provides additional security at the client side as well as at the server side. At the client a new layer of security is to be added over the existing username and password based system compression scheme is used to optimize the size of secret data, first we convert encrypted file in ASCII value it will enable a person to hide minimum 2 times more data in a cover-image, i.e. Now we can store more than Double Amount of Data into Same Image File. This Way we can reduce the cost of data and we can also reduce loss of data minimum 50%. This approach is secure against the RS detection attack and its steno-image is totally indistinguishable from the original image (cover-image) by the human eye And By using Hybrid Approach we can reduce transmission time by Extracting data parallely at decoding time. This way we can reduce time of decoding of message file from the stego-file.

7.Requirement Analysis

7.1 Introduction

7.1.1 Purpose

The proposed system is a product which is capable of hiding the information into other information. The hidden information can be a text file, image file or simply a message. The Carrier file can one of image file. The product (software) should also support cryptography and compression with hybrid approach techniques to improve security level, reduce size of file and also reduce time of decoding(extraction of message file from stego-file) at the receiver end.

7.1.2 Document Conventions

Carrier File/Object: A Carrier file or Carrier object is the source file onto which the data to be hide is written in such a way the output file resembles the initial source file to naked eye and to the normal applications which are associated to the source file. For example, a plain file is generally edited in notepad. If we write some message in some way into the source file and produced an output file, the content of the file should be same as before when the output file is opened in notepad.

Secret File/Object: This object can be message, image file, text file. It is intended to be hidden in the Carrier File.

Hidden File/Object: it is same as Secret File/Object.

Output File: This is the file produced as a result of stenographic operation. It should resemble the Source file.

7.1.3 Intended Audience and Reading Suggestions

End Users: end users are the ones who will be using software the most. This SRS clearly suggests the requirements of the software. So, It is useful to the End Users as well to know what are the requirements and recommendations of the software, although all the requirements are restated into the User Manuals and Installation GUIDE as well.

Developers: The SRS is also useful to the developers who wish to develop their own implantations of Steganography. They can use SRS as directly to enhance the implemented Software or may develop a new Software right from scratch.

Testers: Because the software is bulky and large in size, all the testers testing the Steno-magic software should read SRS carefully to go through all the paths of the software during testing.

Students: The SRS is also knowledgeable for the students to get the information about Steganography. They can refer our SRS, SDD and Project Report to get an overview of Steganography.

7.2. Overall Description: -

7.2.1 Product Perspective

Steganography is a very old technique of hiding the data. This software is all about hiding the data. This software is made according to the modern need of hiding data. It uses various new techniques for hiding the data. The basic advantage of this product is that it is not specific for a particular type of either hidden file or carrier file.

7.2.2 Product Features

- Carrier file can be text file, audio file, video file, and image file
- Various file format is supported
    - Text – txt, rtf, ham, html
    - Image- bmp, gif
- Hidden object can be a text file, video file, audio file, image file
- Hidden files can be of any format
- Encryption is supported
- Compression is supported
- Multiple hidden files can be stored in a single carrier file if the size permit
- Authentication is provided
- Authentication management is also there
- Past steganographic task can be viewed by the help of log file
- Scheduler is also available
- Look of the software can be changed at runtime
- Context sensitive is present
- User manual, installation guide, help file are also provided with the product

7.2.3 User Classes and Characteristics

Administrator: They have full control over the software. Apart from using the basic task, they have full control over the user management and they can also view the log file. End User: They can perform the Steganographic task but have no control over the user management and log file.

7.2.4 Operating Environment

Operating System: Win XP, 2000, 98, Vista
Software req.: JRE 1.6 or higher,
Windows media player Recommended Conf.: 256mb RAM or higher,
10mb Disk space
Screen Resolution: 1024x768

7.2.5 Design and Implementation Constraints

Although java is portable, the software does not run properly on platforms other than windows. It cannot be run over LAN or internet. The software does not check the size limit. As a result, there may be a case that hidden object is not fully consumed yet the software shows no error. This can be verified by extracting the hidden object just after hiding it into the carrier file. Although compression is supported but not implicitly, so manual intervention is needed. Output audio file is distorted considerably when both carrier and hidden objects are audio.

7.2.6 User Documentation

Installation Guide, User Manual and Help File are provided separately with the product. Context sensitive help is integrated implicitly with the product.

7.2.7 Assumptions and Dependencies

Dependencies:

•Mp3plugin.jar is required to play the mp3 files implicitly by the software.
•Windows media player is required to present in the system.
•The Project Location is C:\Package\Stego\. If project is moved to different location the project may not run correctly.

Assumptions:

•The Operating System is any one of NT Family
•Media player is present in the system.
•Class path is set properly
•Mp3plugin.jar is present in the folder C:\Package\Stego

7.3. External Interface Requirements: -

7.3.1 User Interfaces

Splash Screen: It is the first screen which is shown to user. It lodes other modules of the project.
Login Screen: It is for security purpose. It asks the user to give its name and password. Only authenticated user can use the software.
Main Form: It provides various features to user about the mode of use of software. User can choose text, audio, video, image option and can-do work on it. It also provides various other features like user management, compression, encryption, zip, log file.
Help: By clicking on help button help on corresponding topic is shown.
Back: By clicking on this button the main form will appear.
Exit: Click on exit button on main form will shut down the application.
Error Messages: Proper error messages will appear when any error is encountered.

7.3.2 Hardware Interfaces

Apart from the recommended configuration no other specific hardware is required to run the software.

7.3.3 Software Interfaces

The JRE is required to run the software. The JRE version should be 1.6 or higher. The only other software required is the Windows media player to play the video files. The mp3 plugin is also required to play mp3 files.

7.4 Functional Requirements

- Image file is taken as in .jpeg or .gif format.
- The encrypted text file is compressed using LZW compression method.
- Compressed text file is converted into bit stream.
- The bit stream is then embedded into image file using LSB algorithm.
- At the receiver end de-steganography is carried using reverse LSB algorithm.
- The received bit stream is converted into text file then its decompress into encrypted text file.
- Then encrypted text file decrypted.

7.5. Nonfunctional Requirements

7.5.1 Performance Requirements

The ram should be 128mb at least. But 256mb ram is recommended. The disk space required to store the software is 10mb and to store the output files and other configuration files associated with the software the recommended disk space required is 15mb.

7.5.2 Safety Requirements

The size constraints have to be evaluated by the end user only. The software does not check for the size constraint. This is done to enhance the performance of the software in terms of speed. In case size of hidden object exceeds the max. Allowable size that can be hidden, the extra information is truncated. The part of hidden object of size equal to maximum possible size is stored in carrier file. The size of hidden object that can be stored in a carrier file depends on the carrier file size and type of steganographic task. The screen resolution should be set to 1027x768 or higher to get the complete view of the software. In case of lower screen resolution, the software not only looks awkward but also not completely visible.

7.5.3 Security Requirements

The user must have a registered account to run the software. The administrator account is needed tube created on the first run of the software. Then the administrator account can be used to create other

user accounts. If a user forgets the password, the password can be retrieved on the basis of the Date of Birth. This date of birth is stored at the time of creation of the account although it can be modified later.

7.6. Other Requirements

The class path should be set properly. Otherwise the software will automatically set the class path but that will take time, due to which the performance decreases considerably. There are no further requirements other than the specified in this SRS under different headings.

8.Scope

In our randomized LSB technique we have embedded two bits in a pixel using a message dependent randomized approach. In future we would like to exploit the possibility of hiding three bits in moderate bit locations in a randomized manner without disturbing the least significant bit in each pixel. The possibility of embedding by changing every pixel value with a new one which will conceal the hidden data can also be exploited. LSB substitution methods give high capacity. By using better encryption algorithm we can improve the security level and also using better lossless compression algorithm ,we can compress file as much we can.

8.1. Technical Feasibility

It determines the technology needed for the proposed system is available and how this technology can be integrated into the organization. Technical evolution must also assess whether the existing system can be upgraded to use the new technology and whether the organization has the expertise to use it.

8.2. Economical Feasibility

The economic feasibility of the system looks upon the financial aspects of the system. It determines whether is economically feasible or not. In other words, it determines whether the investment that goes into the implementation of the project is recoverable or not. The cost benefit analysis is the commonly used method in evaluating the effectiveness of the system. As the hardware is already available and no investment is to be made in that direction, the only cost involved is that of implementing the system and software.

8.3. Operational Feasibility

It considers whether users will be adapted to the system easily after training, can the system provide them a user friendly environment etc. Acceptance of the system by the user avoiding user resistance is considered as the main factor.

9.Design Details

9.1 Context Level Diagram

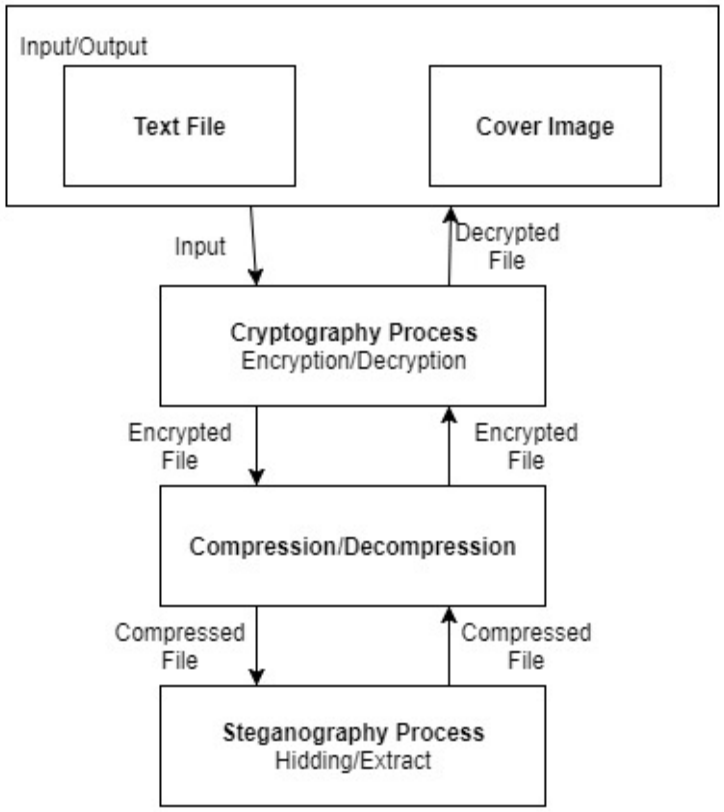

Fig.9.1:Context Level Diagram

9.2 Data Flow Diagram

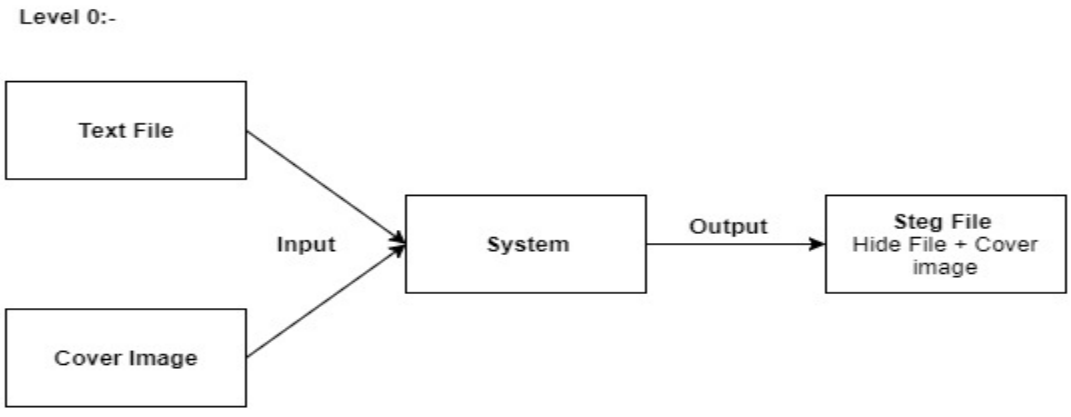

Fig.9.2.1:Level 0 DFD

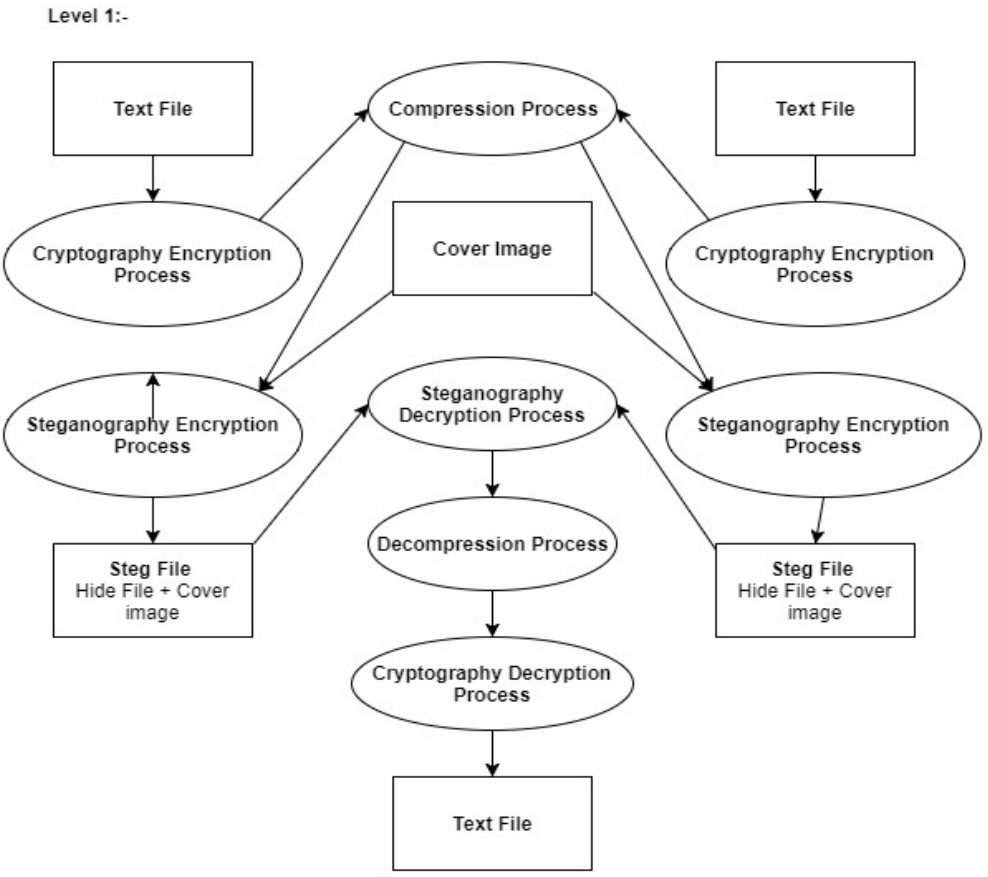

Fig.9.2.2: Level 1 DFD

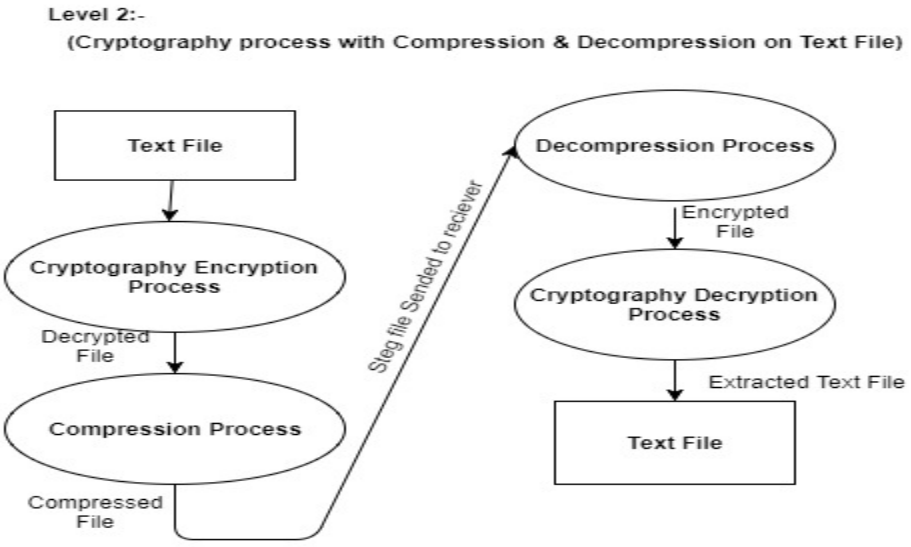

Fig.9.2.3: Level 2 DFD

9.3 Sequence Diagram

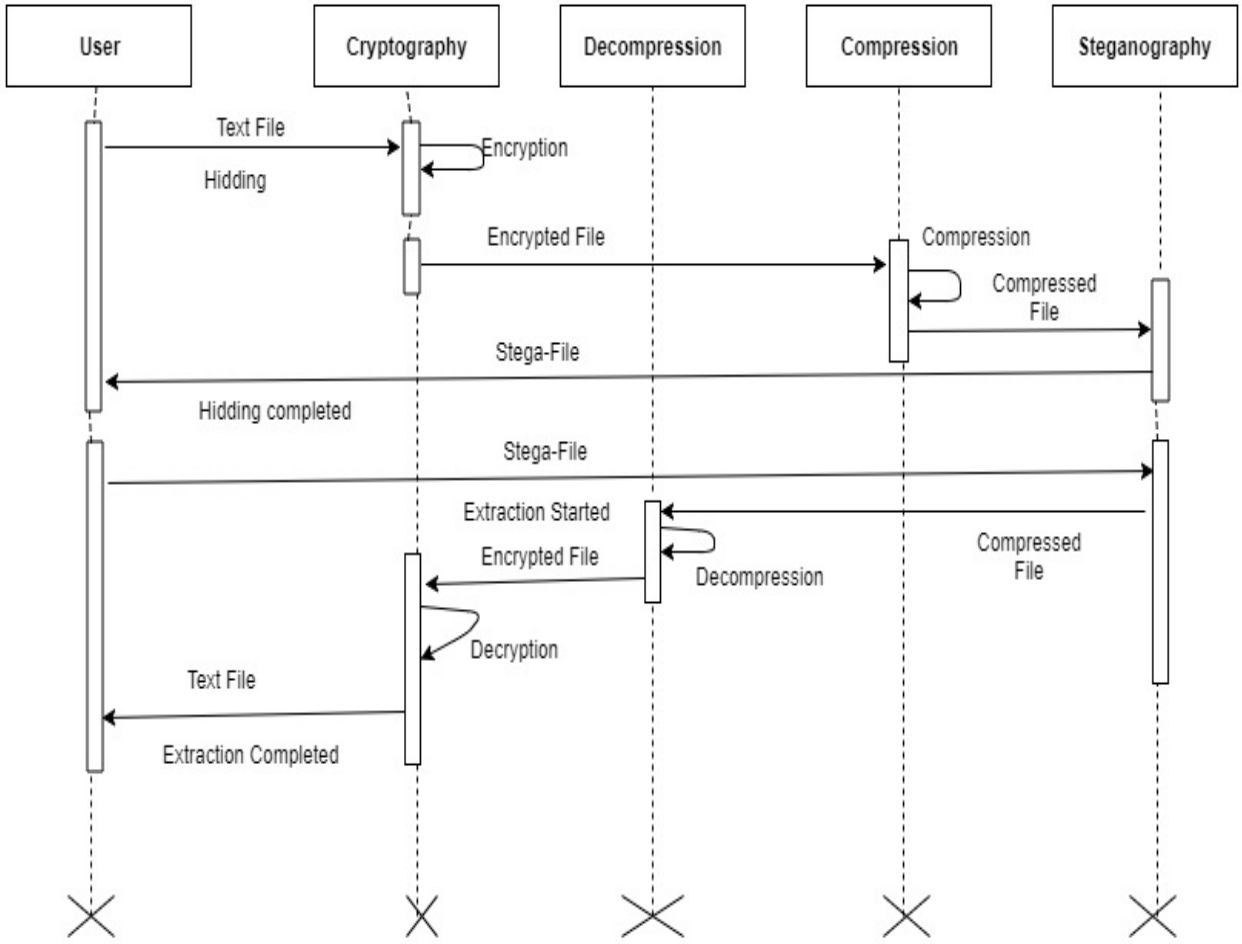

Fig.9.3:Sequence Diagram

9.4 E-R Diagram

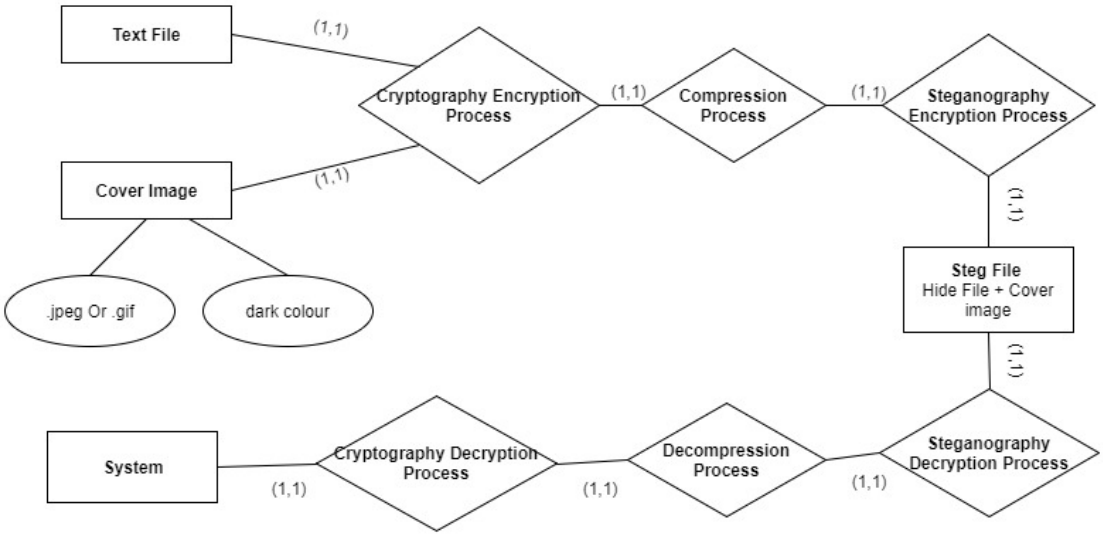

Fig.9.4: E-R Diagram

9.5 Control Flow Diagram

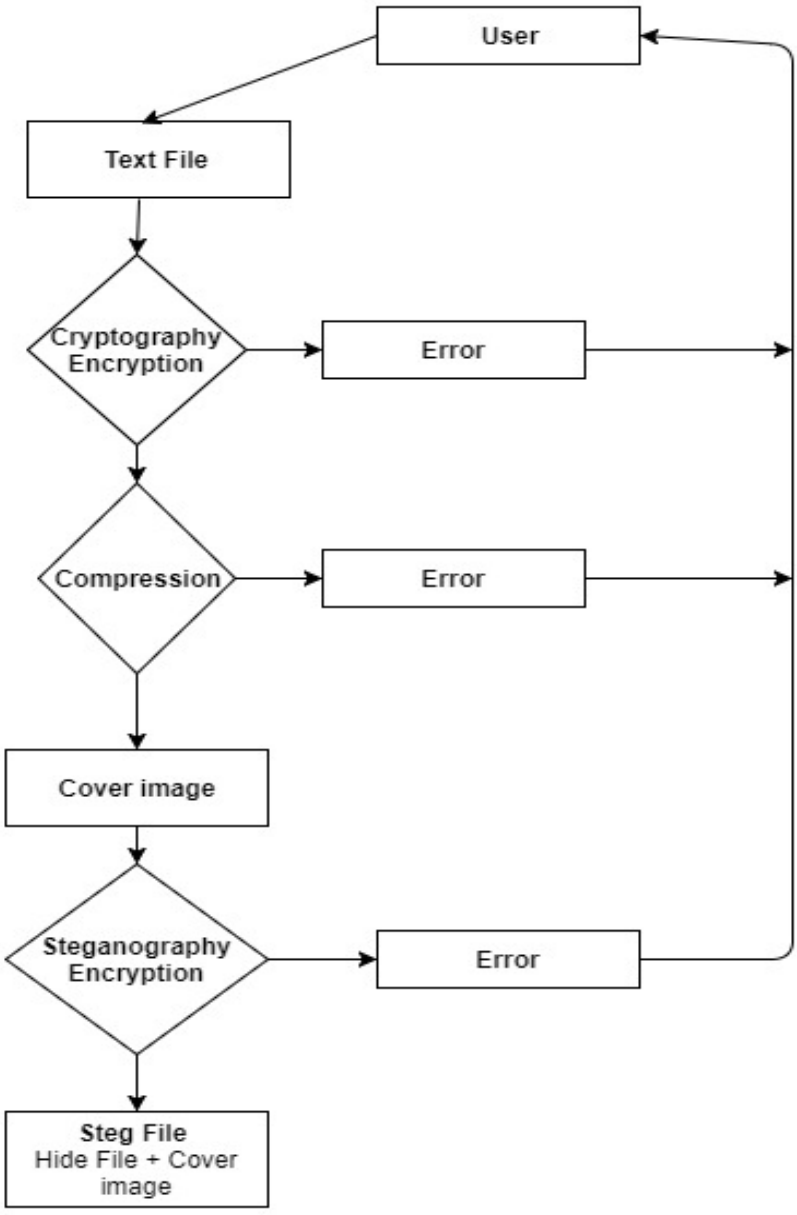

Fig.9.5: Control Flow Diagram

10.Implemention Plan

Our first aim will be to develop our code in different-different modules like first encryption and decryption in cryptography then second steganography, next compression. which will be written in JAVA. Here we will encrypting our text file and then by using LZW method we will compress our file, then we will do stegnography to hide our message into cover image. And at the receiver end first we will decode parallely or extract file from the stego-file by using hybrid technique.

10.1 H/W and S/W requirements

Operating System	:	Windows 7
Application Type	:	Windows App
IDE Platform	:	NetBeans, Eclipse
Coding Language	:	JAVA Programming
Processor	:	Pentium Dual Core
Speed	:	1.65 GHzs(Or more than)
Motherboard	:	Genuine Intel
RAM	:	128Mb(Or more than)
Hard Disk Drive	:	256Mb(Or more than)

10.2 Gantt Chart

10.2.1 Gantt Chart Table

Task Name	Start	End	Duration (days)
Decide The Project Topic	08/01/17	08/06/17	5
Indentify Needs And Constraints	08/07/17	08/12/17	5
Determine Goals And Scope	08/12/17	08/17/17	5
Technical Feasibility	09/01/17	09/06/17	5
Economic Feasibility	09/06/17	09/13/17	7
Application (Code) Feasibili	09/10/17	09/24/17	14
Operation Feasibility	09/20/17	09/30/17	10
Determine Input	10/01/17	10/06/17	5
Determine Output	10/07/17	10/13/17	6
Process Control	10/14/17	10/19/17	5
Synopsis	10/20/17	10/26/17	6

Fig.10.2.1:Gantt Chart Table

10.2.2 Gantt Chart Graph

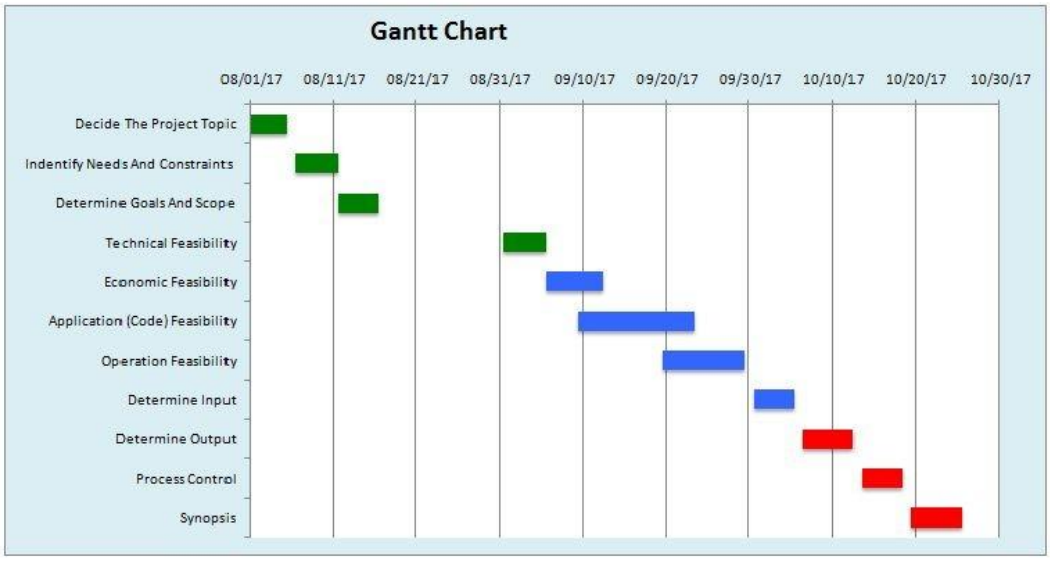

Fig.10.2.2:Gantt Chart Graph

11.Methodology

In this Proposed System, The general scheme for embedding data is depicted in Figure 1. A message is embedded in a file by the stego-system encoder, which has as inputs the original cover, the secret message and a key. The resulting stego-object is then transmitted over a communication channel to the recipient where the stego-system decoder using the same key processes it and the message can be read.

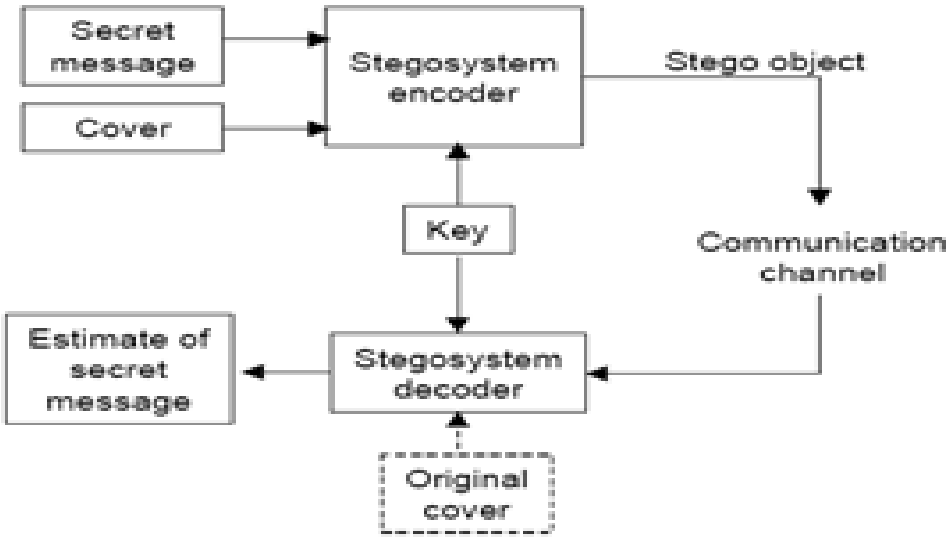

Fig.11.1: A General Steganographic Model

The steganographic process can be represented using formulas. The stego-object is given by: I' = f(I,m,k)
where: I' is the stego-object, I is the original object, m is the message and k is the key that the two parties share. The stego-object may be subject to many distortions, which can be represented as a noise process n:
I'' = I'+n(I')

At the decoder we wish to extract the signal *m*, so we can consider the unwanted signal to be *I*. The embedded signal should resist common signal distortions as those depicted in Figure 2. Two kinds of compression exist: lossy and lossless. Both methods save storage space but have different results. Lossless compression permits exact reconstruction of the original message; therefore it is preferred when the original information must remain intact. Such compression schemes are the images saved as GIF (Graphic Interchange Format). Lossy compression, on the other hand, does not maintain the original's integrity. Such compression scheme is an image saved as JPEG (Joint Photographic Experts Group). The JPEG formats provide close approximations to high-quality digital photos but not an exact duplicate.

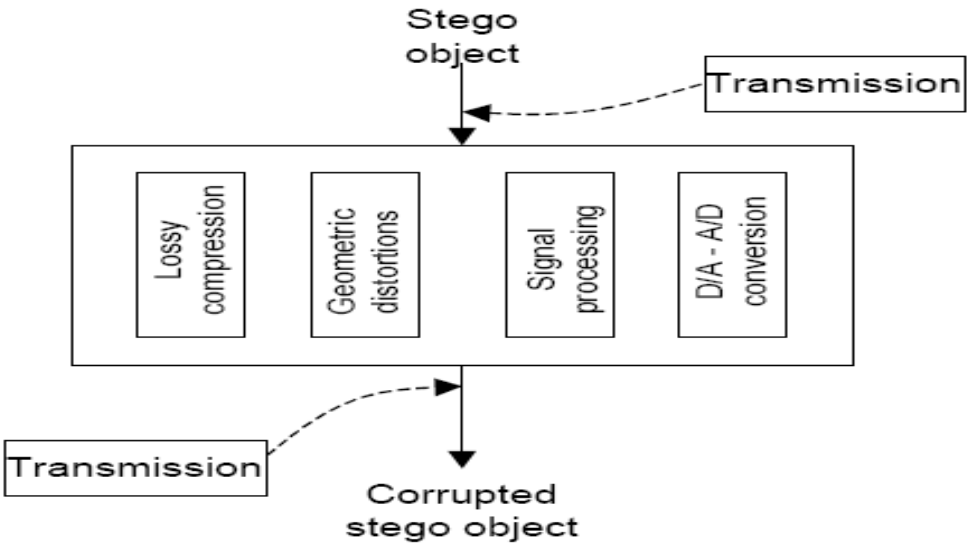

Fig.11.2: Common Signal Distortions over the Transmission Channel

11.1 Lest Significant Bit (LSB)

Least significant bit (LSB) insertion is a common, simple approach to embedding information in a cover image [5]. In this method, we can take the binary representation of the hidden_data and overwrite the LSB of each byte within the cover_image. If we are using 24-bit color, the amount of change will be minimal and indiscernible to the human eye. As an example, suppose that we have three adjacent pixels (nine bytes) with the following RGB encoding:

10010101 00001101 11001001
10010110 00001111 11001010
10011111 00010000 11001011

Now suppose we want to "hide" the following 9 bits of data (the hidden data is usually compressed prior to being hidden): 101101101. If we overlay thes 9 bits over the LSB of the 9 bytes above, we get the following (where bits in bold have been changed):

10010101 0000110**0** 11001001
1001011**1** 0000111**0** 1100101**1**
10011111 00010000 11001011

Note that we have successfully hidden 9 bits but at a cost of only changing 4, or roughly 50%, of the LSBs. Similar methods can be applied to 8-bit palette based images (like GIF images) but the changes, as the reader might imagine, are more dramatic[6]. This is alleviated in this paper by sorting the palette and reassigns each pixel to the index of its color in the new palette before the embedding process.

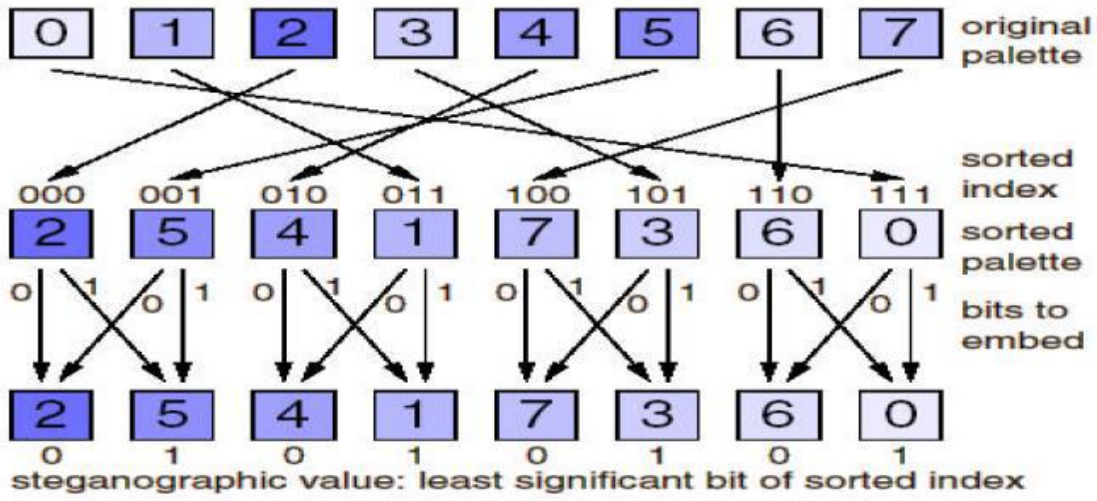

Fig.11.1.1:LSB Diagram

11.2. LEMPEL-ZIV-WELCH (LZW)

LZW is a universal lossless data compression algorithm created by Abraham Lempel, Jacob Ziv, and Terry Welch. It was published by Welch in 1984 as an improved implementation of the LZ78 algorithm published by Lempel and Ziv in 1978. The algorithm is designed to be fast to implement but is not usually optimal because it performs only limited analysis of the data.

The compressor algorithm builds a string translation table from the text being compressed. The string translation table maps fixed-length codes (usually 12-bit) to strings. The string table is initialized with all single-character strings (256 entries in the case of 8-bit characters). As the compressor character-serially examines the text, it stores every unique two-character string into the table as a code/character concatenation, with the code mapping to the corresponding first character. As each two-character string is stored, the first character is sent to the output. Whenever a previously-encountered string is read from the input, the longest such previously-encountered string is determined, and then the code for this string concatenated with the extension character (the next character in the input) is stored in the table. The code for this longest previously-encountered string is output and the extension character is used as the beginning of the next word.

The decompressor algorithm only requires the compressed text as an input, since it can build an identical string table from the compressed text as it is recreating the original text. However, an abnormal case shows up whenever the sequence character/string/character/string/character (with the same character for each character and string for each string) is encountered in the input and *character*/*string* is already stored in the string table. When the decompressor reads the code for *character*/*string*/*character* in the input, it cannot resolve it because it has not yet stored this code in its table. This special case can be dealt with because the decompressor knows that the extension character is the previously-encountered *character*.

A. Compressor Algorithm

Build a table and store all possible strings in it
STRING = get input character
WHILE there are still input characters DO
CHARACTER = get input character
IF STRING+CHARACTER is in the string table then
STRING = STRING+character
ELSE
output the code for STRING
add STRING+CHARACTER to the string table
STRING = CHARACTER
END of IF
END of WHILE
output the code for STRING

B. Decompressor Algorithm

Build a table and store all possible strings in it
Read OLD_CODE
OLD_CODE = get translation of OLD_CODE
output OLD_CODE
CHARACTER = OLD_CODE
WHILE there are still input characters DO
Read NEW_CODE
IF NEW_CODE is not in the string table THEN
STRING = OLD_CODE
STRING = STRING+CHARACTER
ELSE
STRING = get translation of NEW_CODE
END of IF
output STRING
CHARACTER = first character in STRING
add OLD_CODE + CHARACTER to the string table
OLD_CODE = get translation of NEW_CODE
END of WHILE

11.3. THE WORKING SYSTEM ALGORITHMS

11.3.1 The Hiding Algorithm

Input: The text message and cover image represented by animated GIF/JPEG image.
Output: : The stego image represented by animated GIF/JPEG image.
Step 1: Read the text message.
Step 2: Encrypt the text message by using AES algoritwhm.
Step 3: Compress the encrypted message by using LZW compression algorithm.
Step 4: Extract all frames from the stego image(animated GIF/JPEG image) and convert them to a 256 color BMP images.
Step 5: For i=1 to no. of bit in the LZW text code.
Step 6: For j=1 to no. of pixels in each frame.
Step 7: For k=1 to no. of image frames.
Step 8: Hide the LZW code bit in current pixel by using LSB algorithm.
Step 9: next k.
Step 10: next j.
Step 11: next i.

11.3.2 The Extracting Algorithm

Input: The stego image represented by animated GIF/JPEG image.
Output: The text message.
Step 1: Extract all frames from the stego image(animated GIF/JPEG image) and convert them to a 256 color BMP images.
Step 2: Sort the palette of all BMP images and reassign each pixel to its new color index.
Step 3: For i=1 to no. of bit in the LZW text code.
Step 4: For j=1 to no. of pixels in each frame.
Step 5: For k=1 to no. of image frames.
Step 6: Extract the LZW code bit from the current pixel by using LSB algorithm.
Step 7: next k.
Step 8: next j.
Step 9: next i.
Step 10: Decrypt the decompressed text message by using AES algorithm.
Step 11: Decompress the LZW text code by using LZW decompression algorithm.

12.Conclusion

[1]The processing of compressed a text message by using LZW compression method considered as an encryption method, therefore the detection of the hidden message become more complex.
[2] Hide the text in all frames of the animated image and the animation property of the image make the observation of the hidden text very difficult.
[3] The maximum size of the embedded text message could be very huge depend on the LZW code of the text and number of frames in the image.
[4] The lossless compression method used in the image and in the embedded text result to extract the text without any changes in the message.
[5]The processing of hybrid technique to decode message file from stego-file is accomplished.

Acknowledgement

We are greatly pleased in presenting the report on "Efficient data hiding system using LZW, Cryptography and Image Steganography with a hybrid approach". We take this opportunity to express our deep regards towards the ones who offered their valuable guidance in the hour of need. Prominent among them are our guide Prof. Foram Shah and our Head of the Department Prof. Mahendra Patil. The ideas suggested by them are incorporated in this project topic report. Thank you for supporting us and for making our research possible. We express our warm thanks to the Principal, Dr. Shrikant Kallurkar for his support and guidance. Last but not the least we would like to thank our family, especially parents for always putting education first and supporting us in all stages of our life.

Literature Cited

[1] Q. Kester, "A cryptographic Image Encryption technique based on the RGB PIXEL shuffling A cryptographic Image Encryption technique based on the RGB PIXEL shuffling", International Journal of Advanced Research in Computer Engineering & Technology, vol. 2,no.2 pp.848-854,January 2013.

[2] P. Sahute, S. Waghamare, S. Patil, and A. Diwate, " Secure Messaging Using Image Stegnography", International Journal of Modern Trends in Engineering and Research,vol.2,no.3, pp. 598–608, March 2015.

[3] N. Agarwal and P. Agarwal, "An Efficient Shuffling Technique on RGB Pixels for Image Encryption", MIT International Journal of Computer Science & Information Technology, vol. 3, no. 2, pp. 77–81, August 2013.

[4] K. Hamdnaalla, A. Wahaballa, and O. Wahballa, "Digital Image Confidentiality Depends upon Arnold Transformation and RC4 Algorithms",International Journal of Video & Image Processing and Network Security, vol.13, no. 04,August 2013.

[5] N. G. A. P. H. Saptarini, Y. A. Sir, "Digital Color Image Encryption Using RC4 Stream Cipher and Chaotic Logistic Map", Information Systems International Conference, December, pp. 2–4, December 2013.

[6] http://en.wikipedia.org/wiki/RC4 accessed at 25 January 2013.

[7] A. Mousa and A. Hamad, "Evaluation of the RC4 Algorithm for Data Encryption," no. 1, pp. 44–56, June 2006.

[8] B. H. Kamble, "Robustness of RC4 against Differential attack" ,International Journal of computer science and application, vol. 1, no. 4, pp. 661–665, June 2012.

[9] A. M. Abdullah, "New Approaches to Encrypt and Decrypt Data in Image using Cryptography and Steganography Algorithm", International Journal of Computer Applications, vol. 143, no. 4, pp. 11–17, June 2016.

[10] P. R. Deshmukh and B. Rahangdale, "Hash Based Least Significant Bit Technique For Video Steganography", Int. Journal of Engineering Research and Applications, vol.4, no. 1, pp. 44–49, January 2014.