ISSN 2229-5518

Data Hiding Technique using Steganography and Dynamic Video Generation

Abhishek Mangudkar, Prachi Kshirsagar, Vidya Kawatikwar, Umesh Jadhav

Abstract -- Since the dawn of technology, communication has always been in need of novel techniques of data security. Improving the security of data is necessary on a customary basis and with this need we propose our algorithm which is an advanced approach to Steganography. This approach will make sure that the transfer of data becomes more secure against security breaches providing privacy and safe communication environment.

Index Terms - Steganography, Digital Steganography, Digital Video, Data Hiding

1. Introduction

Steganography is an area of information security, where the primary goal is to hide a secret message within a carrier. The carrier can be a message or some other medium, including "overhead" components of an electronic signal. Steganography is the art of concealing messages into something innocuous in such a way that it is extremely difficult for someone to suspect, let alone find, a hidden message. The etymology of the word "Steganography" comes from the Greek language and is translated as steganos-, or "covered", and –graphic, or "writing". So it literally means "concealed writing".

Steganography is often used in an atmosphere of oppression, or when communications and activities must remain secret for fear of reprisal from a watching group or organisation (usually a government). It has also been used extensively in clandestine human source intelligence , where the very existence of a spy, which would be revealed by radio communications, must be concealed.

In the last 40 years, with the advent of personal computing, there has been a rise in digital steganography. Any type of information can be hidden in nearly all files. The best types of file for steganographic transmission are media files due to their large size.

Example:-Text in media files – Text can be embedded in media files by adjusting the file slightly in predefined places so that the difference will correspond to a letter in the alphabet. Pictures can have several specific pixels, a music file some samples, and a video file some of the frames changed a little whilst keeping their functionality majorly intact.

All the standard methods of steganography use an existing file as the 'carrier' file, that is the one in which 'noise' or other factors are added to 'store' or hide data. This pre-required file is not needed in our algorithm as our it will select the file itself from a certain 'images' that ought to be present on both sender and receiver terminals. These images will be used as the frames of the video that will hold the information. Information is also stored in the individual frames using the standard approaches of steganography. Thus the frames and the stored data together will help to retrieve the information from the file. Hence even if the presence of data is detected in the sent video, the random selection of frames that represent partial data will avoid the leakage of the complete information. We have worked on a method for random selection of frames to avoid creating a presence of patterns in the frames. Also the selection is passkey dependant which makes the random selection possible so even if someone hacks the database he will not be aware of the passkey and the security remains intact.

- *Mr. Vidya Kawtikwar, Mrs.Prachi Kshirsagar, Mr. Umesh Jadhav are professors in PVPPCOE, Mumbai, India.*
- *The other mentioned author is currently pursuing a Bachelors degree in Engineering (IT) in PVPPCOE (Class of 2012) University of Mumbai, India.*

E-mail:mangudkarabhishek@gmail.com

2. Related Work

Chameleon [1] is image steganography software developed by Mark David Gan for his thesis at STI College Bacoor, a computer college of the STI Network in the Philippines. Chameleon features a novel adaptive encoding algorithm for 24-bit true-color images based on the steganographic model conceptualized by Yeuan-Kwen Lee and Ling-Hwei Chen1 for greyscale images.

Sujay Narayana and Gaurav Prasad, authors of "Two New Approaches for Secured Image Steganography Using Cryptographic Techniques and Type Conversion' [2] introduced two new methods wherein cryptography and steganography are combined to encrypt the data as well as to hide the encrypted data in another medium so the fact that a message being sent is concealed. One of the methods shows how to secure the image by converting it into cipher text by S-DES algorithm using a secret key and conceal this text in another image by steganographic method. Another method shows a new way of hiding an image in another image by encrypting the image directly by S-DES algorithm using a key image and the data obtained is concealed in another image.

3. Proposed Algorithm Architecture

The proposed architecture is a blend of dynamic video generation and Digital Steganography thus providing a protected and reliable transmission of data over the network. The following diagram represents the working of our proposed algorithm:

The sender and the receiver will possess a database consisting of the 16 same images. Each image will have a 4 bit combination allocated to it. This same 16 images and their associated 4 bit code can be exchanged between the users by meeting face to face or by simply passing it over the net securely.

First input to the proposed algorithm is 4 integer values. The next input by the user will be the data file which is converted into bytes. The whole data is divided into small.

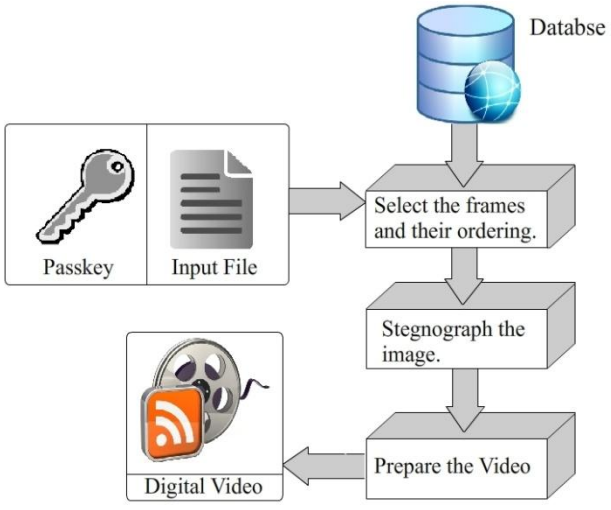

Fig. 1 Proposed Algorithm Architecture

chunks of 4 byte. For the last chunk if the data is less than 4 bytes then dummy data is added to it.

Now the data from the 4 byte chunk is converted into bits resulting in 32 bits of data. Then the 4 bits are selected based on the 4 integer values supplied by the user. The image associated with this 4 bit code is picked.

Now using the key 28 pixels are dynamically selected and the bits are hidden into the respective pixels. Thus each image consists of 4 byte of data hidden in it. The rest chunks are steganograped in the similar manner. Then all the images are combined to form a video which is then passed over the network.

At the receiver end the video file is split back into images. An image comparison algorithm is used to compare the images in the video and find out their respective codes. The bits are placed back in the right position by using the passkey supplied. Even the hidden data in the rest of the image retrieved using the passkey. Thus the data file is reproduced.

4. Proposed System

The algorithm is divided into 3 procedures as given below. We begin by taking the input file 'F' and key 'K' which is of 80 bits(where each 10 bits form a digit).

A. Procedure 1: Frame Selection

Input: File 'F' to be hidden, Key 'K' and Database 'D' consisting of 16 images as stated above
Output: Chunk Array C[] and Image array I[]

Step 1:
Divide the File 'F' into chunk array C[][] where each value in 'C' consist of 4 bytes of 'F'.

Step 2:
Divide passkey 'K' into 4 two digit integer a,b,c,d consisting of consecutive 20 bits from 'K'.

Step 3:
A=a%32
B=b%32
C=c%32
D=d%32
The checksum of a, b, c and d is placed in the 1st pixel of the 1st image. If any of the 4 values are equal than fixed static content is added to the key parts to make them all posses' different unique values.

Step 4:

Generate Code Array 'R []['' where Each Values in R is of 4 bits.

Step5:
Move the bit at C[1][A] into R[1][1].
Move the bit at C[1][B] into R[1][2].
Move the bit at C[1][C] into R[1][3].
Move the bit at C[1][D] into R[1][4].
Repeat this step for all values in C.

Step 6:
Search for Code R in the Database D[] .

Step 7:
Get the Resulting Image into Image array I[]

End Procedure

B. *Procedure 2: Digital Steganography*

Input: Image Array 'I []' and A, B, C, D
Output: Steganographed Image Array I []

Step 1:
Divide each passkey parts A,B,C,D into 2 parts a1,b1,c1,d1,e1,f1,g1,h1 respectively.

Step 2:
Generate Position Array 'P1 []' Using The Following Functions. Let H be the Height of the image and W be the Width of the image
Area=H*W
P1 [1] = (a1*b1+c1)*10%Area
P1 [2] = (a1*c1+b1)*100%Area
P1 [3] = (a1+b1*c1)*150%Area
P1 [4] = (d1*e1+f1)*1000%Area
P1 [5] = (d1*f1+e1)*500%Area
P1 [6] = (d1+e1*f1)*1500%Area
P1 [7] = (g1*h1)*750%Area
P1 [8] = (g1+h1)*2000%Area
P1 [9] = {(a1*b1) + (c1*d1)}-(g1+h1) %Area

The same functions are used again just storing in a2,b2,c2,d2,e2,f2,g2,h2 the 1's complements of a a1,b1,c1,d1,e1,f1,g1,h1.

The same functions are used again once more storing in a3,b3,c3,d3,e3,f3,g3,h3 the EX-OR of a1,b1,c1,d1,e1,f1,g1,h1 and a2,b2,c2,d2,e2,f2,g2,h2.

Thus 27 pixel positions are generated
The 28 Position will be the middle most pixel of the image.
If any of the 28 values are equal than fixed static content is added to the key parts to make them all posses different unique values.

Step 3:
For Each S, Z in Chunk Array C(Consisting of the rest 28 bits) and Position Array P1 respectively
If S==0 Then
I[Z]=I[Z]+1
Else
I[Z]=I[Z]-1
End For

End Procedure

C. *Procedure 3: Creating A Video*

Input: Steganographed Image
Output: Video

Step1:
Take The steganographed Image array 'I []' and convert it into video 'VS' Using Image to Video Conversion Algorithm. The video format of the created video would be '.mov'.

Step2:
Set the basic video properties required to successfully run the video.

End Procedure

At the receiver side the Video is converted back into frames. The checksum of the key entered at the receiver end is verified with the hidden checksum. If they aren't same then no operation is carried out. If receiver enters 3 times a wrong passkey then the video corrupts itself. Then using a Comparison algorithm the appropriate match for the image is found out of the 16 images in the database. The code associated with them is placed at specific location based on the passkey entered, followed by recalculating the position of stored data using the passkey, retrieving the data and thus regenerating the file.

5. Experimental Results

The above algorithm was implemented on JAVA platform for 16 images stored in Oracle Database. The Retrieval algorithm used was 'Content Based Image Retrieval Using Histogram, Entropy and Threshold'. A file 40 bytes was

supplied as an input with the Key K supplied was 12345678. Then a video consisting of 10 frames was formed accordingly. After the process of steganography on the selected images there was no human eye perceptible difference in the resulting image. The fps of video formed was set to 1 and hence a video of 10sec was generated. This was passed through internet to a specified user. The file was completely recovered with no errors when the passkey was correctly entered. If wrong passkey was more than 3 times then the video was corrupted. Thus our proposed algorithm is an effective, reliable and secure one.

6. Conclusion

To summon up creation of the digital video, small size of passkey along with adaptable features to input data are sure features to make this algorithm a very good steganography approach and provide it with very good opportunities. In future this algorithm can be made to hide data in audio, use higher mathematical functions, better validation along with increase in number of frames.

7. Acknowledgment

We sincerely acknowledge the effort and the belief put into our project by our Principal Mr K.T.V Reddy, HOD
Prof. Vaibhav Narawade.

References

[1] Chameleon by David Mark April 2003
[2] Signal & Image Processing : An International Journal(SIPIJ) Vol.1, No.2, December 2010 DOI : 10.5121/sipij.2010.1206 60 titled 'TWO NEW APPROACHES FOR SECURED IMAGE STEGANOGRAPHY USING CRYPTOGRAPHIC TECHNIQUES AND TYPE CONVERSIONS' by Sujay Narayana1and Gaurav Prasad.
[3] Journal of Theoretical and Applied Information Technology © 2005 - 2009 JATIT. All rights reserved. www.jatit.org 35 'A NOVEL INFORMATION HIDING TECHNIQUE FOR SECURITY BY USING IMAGE STEGANOGRAPHY' by M. SITARAM PRASAD 2 S. NAGANJANEYULU 3CH. GOPI KRISHNA 4 C. NAGARAJU.
[4] Lee, Yeuan-Kwen and Ling Hwei Chen. "High Capacity Image Steganographic Model." Vision, Image and Signal Processing. IEEE Proceedings 147.3 (2000): 288-294.
[5] Khan, David. "The History of Steganography." Information Hiding: First International Workshop. Lecture Notes in Computer Science 1174 (1996): 1-5.
[6] Anderson, Ross J. and Fabien A. P. Petitcolas. "On The Limits of Steganography." Special Issue on Copyright & Privacy Protection. IEEE Journal of Selected Areas in Communications 16.4 (1998): 474-481.
[7] Lee, Yeuan-Kwen and Ling Hwei Chen. "An Adaptive Image Steganographic Model Based on Minimum-Error LSB Replacement." Ninth National Conference on Information Security. (1999): 8-15.
[8] Pfaffenberger, Bryan and David Wall. Que's Computer and Internet Dictionary. 6th ed. Indianapolis: Que, 1995.
[9] Jamil, T., "Steganography: The art of hiding information is plain sight", IEEE Potentials, 18:01, 1999
[10] Wang, H & Wang, S, "Cyber warfare: Steganography vs. Steganalysis", Communications of the ACM,47:10, October 2004 [5] Anderson, R.J. & Petitcolas, F.A.P., "On the limits of steganography", IEEE Journal of selected Areas in Communications, May 1998
[11] Marvel, L.M., Boncelet Jr., C.G. & Retter, C., "Spread Spectrum Steganography", IEEE Transactions on image processing, :08, 1999
[12] Dunbar, B., "Steganographic techniques and their use in an Open-Systems environment", SANS Institute, January 2002 [8] Artz, D., "Digital Steganography: Hiding
[13] Data within Data", IEEE Internet Computing Journal, June 2001 Simmons, G., "The prisoners problem and the subliminal channel", CRYPTO, 1983
[14] Chandramouli, R., Kharrazi, M. & Memon, N., "Image steganography and steganalysis: Concepts and Practice", Proceedings of the 2 nd International Workshop on Digital Watermarking, October 2003