Secure Data Transmission Using Video Steganography

R. Balaji
Computer Science and Engineering department,
Sri Sai Ram Engineering College,
Chennai, India.
balajirajagopal@live.in

G. Naveen
Dept. of Information Science and Technology,
College of Engineering, Anna University,
Guindy, Chennai, India.
naveenganapathi@gmail.com

Abstract – It is very essential to transmit important data like banking and military information in a secure manner. Video Steganography is the process of hiding some secret information inside a video. The addition of this information to the video is not recognizable by the human eye as the change of a pixel color is negligible. This paper aims to provide an efficient and a secure method for video Steganography. The proposed method creates an index for the secret information and the index is placed in a frame of the video itself. With the help of this index, the frames containing the secret information are located. Hence, during the extraction process, instead of analyzing the entire video, the frames containing the secret data are analyzed with the help of index at the receiving end. When steganographed by this method, the probability of finding the hidden information by an attacker is lesser when compared to the normal method of hiding information frame-by-frame in a sequential manner. It also reduces the computational time taken for the extraction process.

Keywords – Video Steganography, Video Frame, Index, LSB, Secret Data

I. INTRODUCTION

Data hiding techniques have been used widely for the transmission of data over a long time. They are classified into two types: Watermarking and Steganography. Steganography comes from the Greek word “steganos” and “graptos” meaning covering and writing respectively [1]. It is the art of embedding a message that is to be hidden in a medium, usually a picture, an audio file, or a video file, in such a way that no one apart from the sender and the intended recipient even realizes that there is a hidden message.

Fig.1 describes the basic Steganography and De – Steganography process. The newest form of Steganography has become the target of researchers to find new ways to embed hidden messages of larger sizes. Steganography on video files answer these needs for larger spaces in hiding data. Larger space for embedding and having small unnoticeable distortions make video Steganography a reliable method in hiding data.

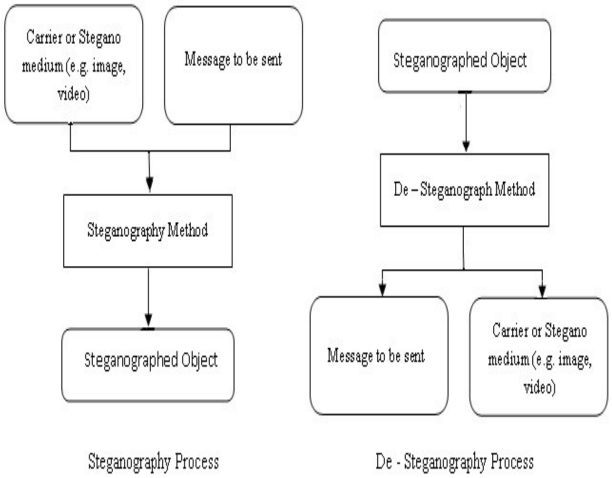

Fig.1 Steganography and De – Steganography Process

This paper describes a simple method of embedding secret data into a video.

II. RELATED WORKS

The techniques involved in Steganography have been divided into five categories [2] [3]. The first category is the Spatial Domain-based Steganography. It consists of the Least-Significant Bit (LSB) Replacement and Matching and the BPCS methods. The second is the Transform Domain-based Steganography where programs such as JSteg, F3, F4, and F5 algorithms are used. The third category is Document-based Steganography. Data are embedded in document files by adding tabs or spaces to .txt and.doc files. File Structure-based Steganography is the fourth category where this time, structural embedding is used to insert secret data in the redundant bits of the file such as the reserved bits in the file header or the marker segments in the file format. The last category of Steganography includes the few methods that are based on video compression encoding and spread spectrum technique [4].

Recent Video Steganography techniques use swapping algorithm and UTF-32 coding scheme to embed data into a video. The method works in a manner of hiding the data first in an image file and it will then be attached to a cover media which is the video file in AVI format. The hiding technique of the image file in the AVI file is done by using one of the four methods: RGB (Red Green Blue), Discrete Fourier Transform, Scalar Costa Scheme, and the Junk Replacement Method. [5].

III. OVERVIEW

The process of retrieving the secret data from the steganographed video involves searching the entire video for the secret data, which increases the computational time and this adds an overhead to the extraction process. This paper provides a solution to the problem by creating an index for the secret data which is stored in the video. Just like the index of a book, the index for the secret data helps to find the position of the secret data in the video without analyzing the entire video. The index for the secret data is determined by a set a mathematical functions which may depend on the characteristics of the video.

IV. PROPOSED SYSTEM

The method we propose here effectively hides the secret data into a video using the existing Steganographic techniques. A video file is usually composed of several frames [6]. This method uses some frames (or images) of the video to hide the secret message. The secret data is not hidden in sequential frames. Rather, they are placed in random frames. Hence each frame that contains the secret data can be identified using the index. The Index Frame, which is placed in some frame of the video, provides the information about the frames that contain the secret data. The remaining frames in the video which do not contain secret data are also steganographed with some random data. This provides additional security to the secret data. The proposed system consists of three phases:

1. Analyzing the video.
2. Determination of index frame and its data.
3. Determination of frames for secret data.

The process of extracting the secret data from the steganographed video is just the reverse process of hiding the secret data into a video. The steganographic technique we used here is based on LSB (Least Significant Bit Insertion) method.

V. IMPLEMENTATION

1. *Analyzing the video*

Consider a video of '$\mathbf{y}$' seconds with '$\mathbf{x}$' frames per second. Hence, the total number of frames in that video is '$\mathbf{f} = \mathbf{x} \times \mathbf{y}$'. Let the resolution of the video be ‘$\mathbf{m} \times \mathbf{n}$‘. In the entire video, only certain frames are used for storing the secret data. The rest of the frames are filled with some random data to provide security to the secret data. The ratio of the frames to be used for storing the hidden information to the frames to be used to contain the random data is taken as ‘$\mathbf{b:c}$‘. Now, the frames to be used for Steganography is '$\mathbf{u} = (\mathbf{b}/(\mathbf{b}+\mathbf{c})) \times \mathbf{f}$‘.

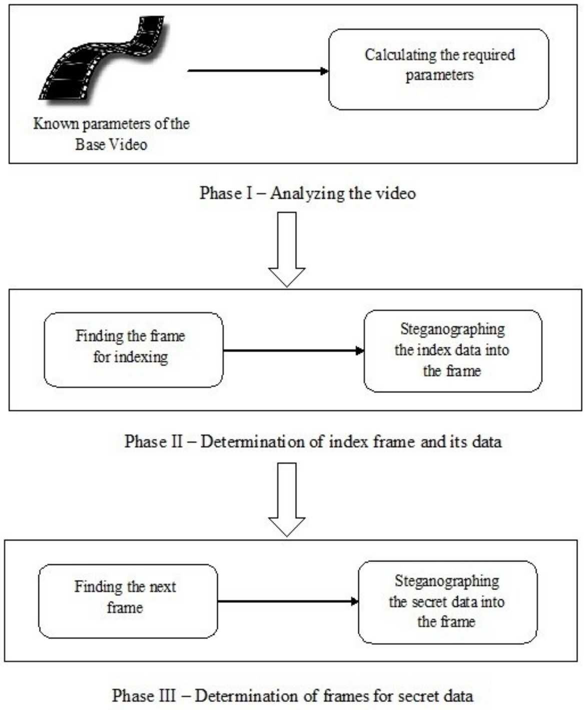
Calculating the required parameters
Known parameters of the Base Video
Phase I – Analyzing the video
Finding the frame for indexing
Steganographing the index data into the frame
Phase II – Determination of index frame and its data
Finding the next frame
Steganographing the secret data into the frame
Phase III – Determination of frames for secret data

Fig.2 Implementation of Proposed System

Let the size of the data (to be hidden) be '$\mathbf{t}$' bits. The video consists of ‘$\mathbf{x} \times \mathbf{y} \times \mathbf{m} \times \mathbf{n} \times \mathbf{24}$‘ bits (assuming that each pixel has 24 bits [RGB]). One frame is used as the Index Frame ‘$\mathbf{i}$’ and ‘$\mathbf{(u-1)}$‘ frames are used to hide the 't' data bits. Hence, ‘$\mathbf{ceil(t/(u-1))}$‘ bits of secret data should be placed in each of the (u-1) frame. The spacing between these frames in the entire video is given as '$\mathbf{z} = \mathbf{floor\ (f/u)}$‘. The '$\mathbf{v}$'th frame is determined by ‘$\mathbf{(i+v \times z) \bmod f}$‘.

Each frame has $\mathbf{m} \times \mathbf{n} \times \mathbf{24}$ bits of information. The number of significant bits of a pixel that should be replaced to fit in the data to be hidden is the minimum value of '$\mathbf{k}$' for which $\mathbf{m} \times \mathbf{n} \times \mathbf{k}$ is just greater than '$\mathbf{ceil(t/(u-1))}$' which is the size of the data(in bits) to be hidden in each frame. The unused frames are filled with ceil(t/(u-1)) random bits to provide security to the secret data.

2. *Determination of Index Frame and its data*

The Index Frame is placed in some frame of the video and its position is obtained by applying a function over the parameters of the video. The position of the index frame is obtained by ‘$\mathbf{i} = \mathbf{function1\ (parameters\ of\ video) \bmod f}$‘. Within this 'i'th frame, the index data is stored in the least significant '$\mathbf{j}$' bits of certain pixels where 'j' is the minimum value for which $\mathbf{m} \times \mathbf{n} \times \mathbf{j}$ is just greater than '$\mathbf{p}$' where 'p' is the size of the index frame. The structure of Index Frame is described in Fig. 3.

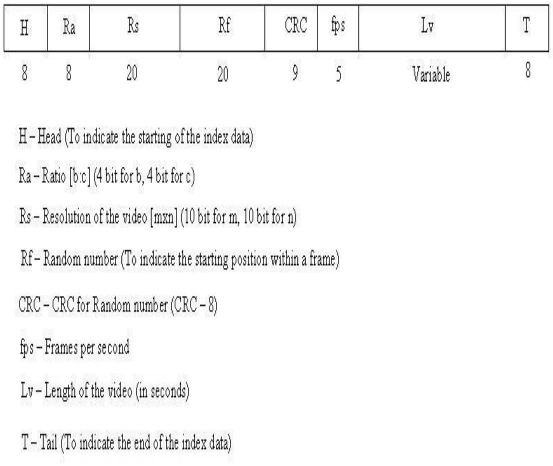
H
Ra
Rs
Rf
CRC
fps
Lv
T
8
8
20
20
9
5
Variable
8
H – Head (To indicate the starting of the index data)
Ra – Ratio [b:c] (4 bit for b, 4 bit for c)
Rs – Resolution of the video [mxn] (10 bit for m, 10 bit for n)
Rf – Random number (To indicate the starting position within a frame)
CRC – CRC for Random number (CRC – 8)
fps – Frames per second
Lv – Length of the video (in seconds)
T – Tail (To indicate the end of the index data)

Fig.3 Structure of Index Data

The starting pixel which contains the index data is given by ‘**s = function2 (parameters of video) mod (m×n)**‘. From there, the entire index data is placed at equidistant places in the frame. This equidistance is obtained as '**l=floor((m×n)/p)**‘. The '**h**'th place where the index data is present is given by **(s+(h-1)×l) mod (m×n)**. Thus the index data is placed in Index Frame. The same procedure is followed to obtain the entire index data during the process of de – Steganography. The functions ‘function1’ and ‘function2’, which are used above, can yield a value by performing any mathematical operations on the parameters passed.

3. *Determination of frames for secret data*

After placing the index data, the secret data has to be inserted in some frames of the video. The frames which contain the secret data are **(i+z) mod f, (i+2×z) mod f, (i+3×z) mod f and so on**. Each frame has '**g = ceil(t/(u-1))**’ bits of secret data in it. The starting position of this secret data within this frame is given by the random number. Let '**d**' be the starting pixel which contains the secret data. We get **d = (random number) mod (m×n)**. Within this frame, the secret data is placed equidistant. The distance is given by '**q=floor((m×n)/g)**’. Hence, the hidden data is placed at **d, (d+q) mod (m×n), (d+2×q) mod (m×n), (d+3×q) mod (m×n) and so on**. The starting and the ending of the secret data may be padded with some bits in order to identify the correctness of the enclosed secret data.

VI. ALGORITHM

1. Obtain the inputs - video, data to be hidden, ratio of used to unused frames, random number.

2. Analyze the given inputs and find the required parameters

Variables	Description
y	Length of video in seconds
x	Frames per second
m×n	Resolution of the video
D	Random number
F	Total number of frames in the video
U	Frames to be used
T	Size of secret data
I	Index Frame in the video
ceil (t/(u-1))	Number of secret data bits in each of the (u-1) frames
Z	Spacing between the used frames
(i+v×z) mod f	‘v’th frame which contains the hidden data
K	Least significant bits of the pixel to be replaced with the index data in the index frame
j	Least significant bits of the pixel to be replaced with the secret data in the (u-1) frames
l	Distance between two pixels which contains index data
q	Distance between two pixels which contains the secret data
s	Starting pixel which contains index data in index frame
d	Starting pixel which contains secret data in (u-1) frames

Table.1 Parameters to analyze and find

3. Compute i = function1(parameters of video) mod f, s = function2(parameters of video) mod (m×n)

4. In Index Frame, replace the least significant 'k' bits of s, (s+l) mod (m×n), (s+2×l) mod (m×n) and so on till the index data is placed in the frame.

5. Calculate the subsequent frames as (i+z) mod f, (i+2×z) mod f, (i+3×z) mod f and so on.

6. In each of the above frames, replace the least significant 'j' bits of the pixels d, (d+q) mod (m×n), (d+2×q) mod (m×n), (d+3×q) mod (m×n) and so on until all the hidden data for that frame which is ceil(t/(u-1)) bits are placed.

7. In the unused frames, replace the least significant 'j' bits of the pixels d, (d+q) mod (m×n), (d+2×q) mod (m×n), (d+3×q) mod (m×n) and so on with some random data.

VII. RESULT AND OBSERVATION

Due to difficulty of showing the result as a video stream on paper, we thought of displaying the result on the frame of the digital video file along with its histogram. Fig. 4 shows the frame before applying the algorithm, while Fig. 5 shows the same frame after applying the algorithm.

We observe that there is so significant difference (both images resemble the same to human eye) between the two frames. This shows that the algorithm can be applied successfully on video frames. The next step was to verify the algorithm through histogram, to see the divergences on the frames before and after hiding data in it.

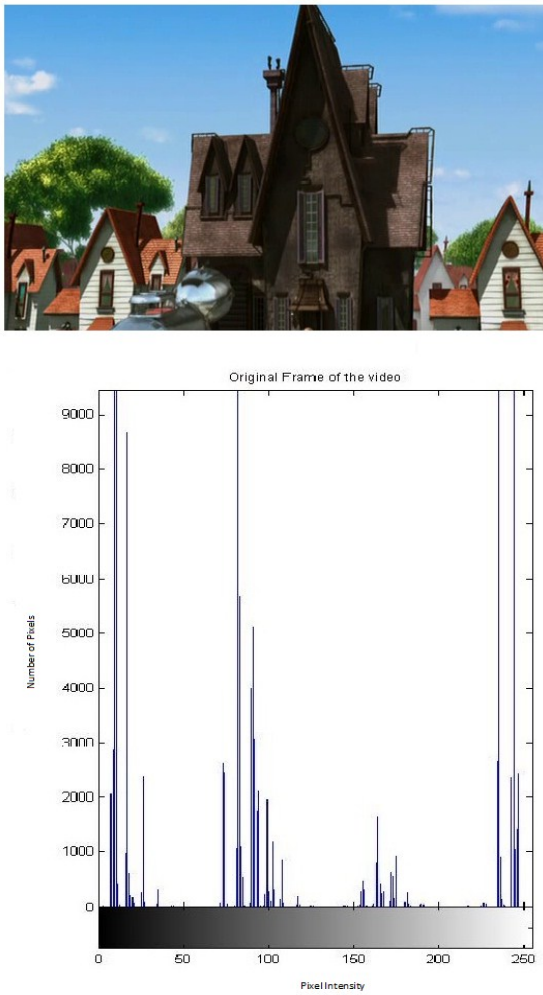

Fig.4 Frame of the video without hiding data and its histogram

From the histogram for both Fig.4 & 5 of the frames, the difference between the two frames before and after hiding the data can be clearly seen, which prove that the algorithm successfully hid the data into the frames without making noticeable difference for the human vision.

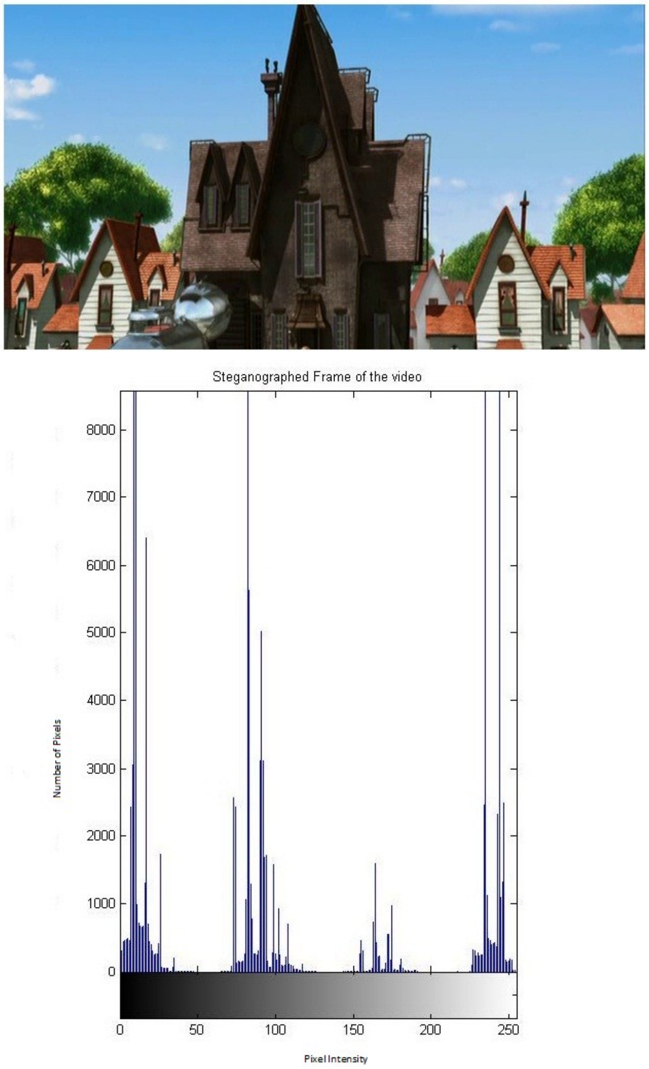

Fig.5 Frame of the video with hiding data and its histogram

Another important observation in the result is that it is good to replace only a few least significant bits of the pixel so that it doesn't affect the quality of the video. Hence, the values of 'k' and 'j' should be kept as minimum as possible. For this, the length of the video must increase as the size of the secret data increases.

VIII. ADVANTAGES

1. Less computational time

Since the proposed system uses indexing concept, the process of retrieving the secret data from the steganographed video becomes very simple and requires very less time.

2. Highly secure

Since random data are also placed in unused frames in the video, the attacker is left clueless to know the real secret data hidden in the video. Hence highly confidential data like military secrets and bank account details can be easily steganographed in ordinary video and can be transmitted over internet even in unsecured connection.

IX. FUTURE WORK

In the proposed method, the position of the frames and the position of the pixels within a frame which contain the secret data are placed equidistant. An algorithm, which can determine these positions, can be developed. This will further enhance this method of Video Steganography.

X. CONCLUSION

Thus, this paper provides a feasible solution for Video Steganography. The method proposed here considers video as set of frames or images and any changes in the output image by hidden data is not visually recognizable. It also makes use of simple mathematical calculations which bring down the computational time very less making it a very simple and effective method for video Steganography. Apart from its simplicity in implementing, it also provides security for the secret data in the transmission.

REFERENCE

[1] http://en.wikipedia.org/wiki/Steganography

[2] Ming, Chen, Zhang Ru, Niu Xinxin, and Yang Yixian, “Analysis of Current Steganography Tools: Classifications & Features,” Intelligent Information Hiding and Multimedia Signal Processing, 2006. IIHMSP '06. International Conference on Dec. 2006 Page(s):384 – 387.

[3] Mehdi Kharrazi, Husrev T. Sencar and Nasir Menon, "Image Steganography: concepts and practice," Lecture notes on Computer science, vol.2939, 2004, pp.204-211

[4] Shirali - Shahriza, "A new method for real - time Steganography: 8th International Conference on Signal Processing, vol. 4, 2006, pp. 16-20.

[5] Kavitha, R. and A Murugan, “Lossless Steganography on AVI File using Swapping Algorithm,” SRM University. Conference on Computational Intelligence and Multimedia Applications, 2007. International Conference on Volume 4, 13-15 Dec. 2007 Page(s):83 – 88.

[6] G. Doerr and J. Dugelay, "Security pitfalls of frame-by-frame approaches to video watermarking", IEEE Transactions on Signal Processing, vol. 52, 2004, pp. 2955-2964