Secure Video Communication For Defense Applications

[1]Madiraju Sirisha, [2]Parusharamu Easari
[1,2] Assistant Professor
[1]Department of Electronics and Communications Engineering,
[1]Nalla Malla Reddy Engineering College, Divya Nagar, Ghatkesar Mandal, RR Dist., Hyderabad, India
[2]Sri Indu College of Engineering and Technology, Sheriguda, Hyderabad, India

Abstract : The aim of this review is to use the video file as a cover carrier in the study of steganography methods. A video file consisting of separated images and audio files which is video based steganography is used more as it is more eligible than other multimedia files. The advantages of using the video file as a cover carrier for steganography has been proposed as a result of this study. .

Hiding confidential information within any media is called as Steganography technique. Since Steganography and cryptography are similar in the way that they both are used to protect confidential information, there is a confusion between them. The difference between the two is in the appearance in the processed output; the output of cryptography draws the attention since it is scrambled whereas in Steganography the output is not apparently visible. In this article we have tried to elucidate the different approaches towards implementation of using 'multimedia' file (text, static image, audio and video) and Network IP datagram as cover in steganography. Some steganalysis methods will also be discussed.

IndexTerms **– video processing, steganography, cryptography**

I. INTRODUCTION

There is a large quantity of literature in hiding of data and watermarking in digital image and raw video. This paper focuses the internal dynamics of video compression, particularly the motion estimation stage. This stage is chosen as its contents are processed internally during the video encoding decoding which makes it hard to be detected by image steganalysis methods and is lossless coded, so that quantization distortions does not take place. In motion vectors the data hiding is done by changing them with the properties such as amplitude, phase angle etc.

The message data bits are hidden in some of the motion vectors whose magnitude is above a predefined threshold called as candidate motion vectors (CMVs). A single bit is hidden in the least significant bit of the larger component of each CMV, the data is encoded as a region where the motion estimation is only allowed to generate motion vectors in that specified region.

The authors embed the data in video using the phase angle between two consecutive CMV. Selection of these CMV is based on the magnitude of the motion vectors. The phase angle difference in sectors between CMV is encoded by the message bit stream. The block matching is constrained to search within the selected sector for a magnitude to be larger than the predefined threshold. The methods are focused on finding a direct reversible way to identify the CMV at the decoder and thus relied on the attributes of the motion vectors. In this paper, we take a different approach directed towards achieving a minimum distortion to the prediction error and the data size overhead. This approach is based on the associated prediction error and we are faced by the difficulty of dealing with the nonlinear quantization process.

Steganography is the art and science of writing hidden messages in such a way that no one apart from the sender and intended recipient, suspects the existence of the message, a form of obscurity. The main advantage of steganography over cryptography is that messages in cryptography do not attract attention to themselves. Plainly visible encrypted messages no matter how unbreakable will arouse suspicion, and may in themselves be incriminating in countries where encryption is illegal. Therefore, steganography protects both messages and communicating parties whereas cryptography contents of a message.

Steganography includes the hiding of information within computer files. Electronic communications include steganographic coding inside transport layer, such as a document file, image file, program or protocol. Ideal files for steganographic transmission are media files because of their large size. As a simple example, a sender might start with an innocuous image file and adjust the color of every 100th pixel to correspond to a letter in the alphabet, a change is so difficult to understand that someone not specifically looking for it is unlikely to notice it.

A digital watermark is a marker secretly embedded in a noise-tolerant signal such as audio or image data. The ownership copyright of such signal can be identified. "Watermarking" is the process of hiding digital information in a carrier signal; the hidden information should, but does not need to contain a relation to the carrier signal. To verify the genuineness or integrity of the carrier signal or to show the identity of its owners, digital watermarks are used. It is mainly used for tracing copyright infringements and for banknote authentication. Like traditional watermarks, digital watermarks are only capable of being perceived under certain conditions, i.e. after using some algorithm, and not perceived anytime else. It is of no use if a digital watermark distorts the carrier signal in a way that it becomes perceivable. Traditional Watermarks are applicable to visible media (like images or video), whereas in digital watermarking, the signal may be audio, pictures, video, texts or 3D models. At a time a signal may carry several different watermarks. A digital watermark does not change the size of the carrier signal unlike metadata.

Audio Steganography based on LSB: If a audio file with ".wav" extension has been selected as host file then it is assumed that the least significant bits of that file should be modified without degrading the sound quality.

Video Steganography based on LSB: If a video file with ".avi" extension has been selected as host file then It is assumed that the least Significant bits of that file should be modified without degrading the image quality.

II. DESCRIPTION

BLOCK DIAGRAM

EMBEDDING PART

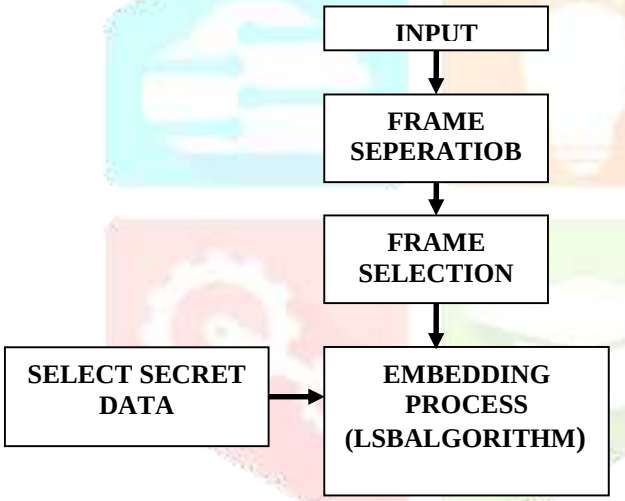

Fig.1. Embedding process

EXTRACTING PART

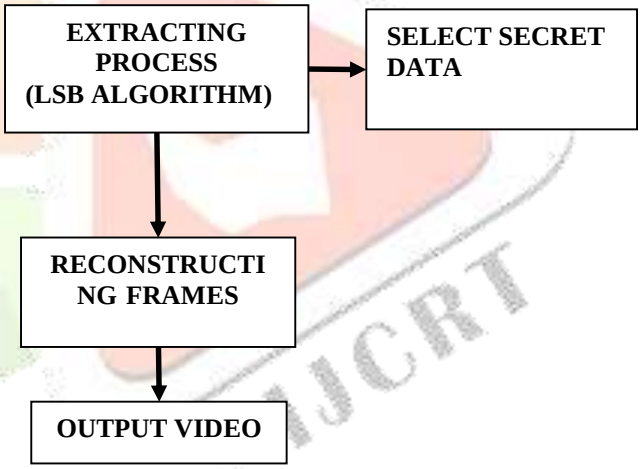

Fig.2. Extracting process

Process of selecting a video: Digital video relates to the storage, manipulation, and capturing of moving images that can be displaced on computer screens. The moving images should be digitally handled by the computer. Digital is referred to a discontinuous events opposite to analog which is a continuous event.

Camera and microphone captures the picture and sound of a video session and sends analog signals to a video-capture adapter board. Only half of the number of frames per second are captured by the board in order to reduce the amount of data to be processed. Further, there is an analog-to-digital converter chip on the video-capture adapter card which converts the analog signals to digital patterns (0s and 1s). Next, a compression/decompression chip or software reduces the data to a minimum necessary for recreating the video signals. This process is more efficient as no analog is involved.

Easy to manipulate: Comparison of a typewriter with a word processor is an example to show The difference between analog and digital. Like a cut and paste function with a word processor, editing is easier and faster with a digital video. Also, many effects that were exclusive for specialized post production houses are now easily achieved by bringing files from Flash, Photoshop and Sound Edit as components in a video mix. Separation of sound from image enables editing one easy without disturbing other.

Data Preservation: There are so many big screen films which maintain good quality though they are not digital but it is easy to maintain the quality of video in digital. It is easy to maintain DVD or hard disks rather than traditional tapes as they subjects to wear and tear more. Also, Analog signals easily distorts and will lose much of the original data after a few transfers where as digital doesn't.

Frame Separation: Frame processing is the first step in the background subtraction algorithm, the purpose of this step is to prepare the modified video frames by removing noise and unwanted object's in the frame so that the amount of information gained from the frame is increased and the sensitivity of the algorithm.

In preprocessing the raw input video info is changed by collecting simple image processing tasks. This can be done by subsequent steps. Preprocessing of the video is necessary to improve the detection of moving object's For example, by spatial and temporal smoothing, snow as moving leaves on a tree, can be removed by morphological process of the frames after identifying the moving object's as shown in fig. 3

Fig. 3 Preprocessing of the Video

Another key issue in pre processing is the data format used by the particular background subtraction algorithm. Most of the algorithm handles luminance intensity, which is one scalar value per each pixel in either RGB or HSV color space as it is becoming more popular in the background subtraction algorithms.

Coding for Frame Separation

```
file=aviinfo('movie1.avi');
frm_cnt=file.NumFrames
str2='.bmp'
   h = waitbar(0,'Please wait...');
for i=1:frm_cnt
   frm(i)=aviread(filename,i);
   frm_name=frame2im(frm(i));
   frm_name=rgb2gray(frm_name);
   filename1=strcat(strcat(num2str(i)),str2);
   imwrite(frm_name,filename1);
   waitbar(i/frm_cnt,h)
end
close(h)
```

Separation logic gives the following reasons:

1. Pointer data structures, manipulation programs — including hide of information in the presence of pointers;
2. "transfer of ownership" (avoidance of semantic frame axioms); and
3. Virtual separation (modular reasoning) between concurrent modules.

Separation logic supports the developing field of research described by Peter O'Hearn and others as local reasoning, whereby specifications and proofs of a program component mention only the component used memory, and not the total system global state. Automated program verification (where an algorithm checks the validity of another algorithm) and automated parallelization of software are the applications.

Digital image fundamentals: Digital image is defined as a two dimensional function f(x, y), where x and y are spatial (plane) coordinates, and the amplitude of f at any pair of coordinates (x, y) is called intensity or grey level of the image at that point. The field of digital image processing refers to processing digital images by means of a digital computer. The digital image is composed of a finite number of elements referred as image elements, picture elements and pixels, each of which has a particular location and value. Digital

Image compression addresses the problem of reducing the amount of data required to represent a digital image. The basic of the reduction process is removal of redundant (repeated) data. From the mathematical viewpoint, this amounts to transforming a 2D pixel array into a statically uncorrelated data set. The data redundancy is a mathematically quantifiable entity. If n1 and n2 denote the number of information-carrying units in two data sets that represent the same information, the relative data redundancy R_D [2] of the first data set (the one characterized by n1) can be defined as $R_D = 1 - \frac{1}{C_R}$

Where C_R called as compression ratio [2]. It is defined as $C_R = \frac{n1}{n2}$

In image compression, three basic data redundancies can be identified and exploited: Coding redundancy, interpixel redundancy, and phychovisal redundancy. When one or more of these redundancies are eliminated, image compression is said to be achieved.

Image Compression Model:

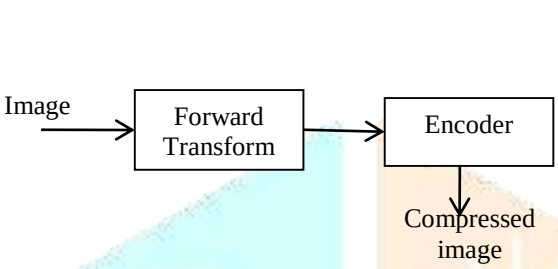

Fig. 4.a. Block Diagram of Image compression

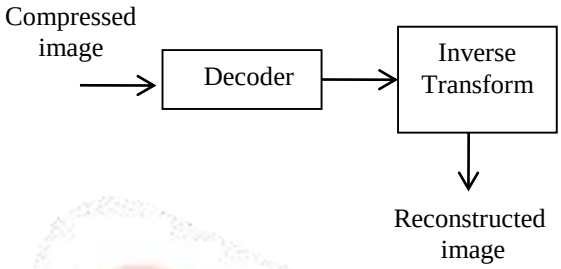

Fig.4.b Block Diagram of Image Decompression

There are many methods available for loss and lossless, image compression. The efficiency of these coding standardized by some Organizations. The International Standardization Organization (ISO) and Consultative Committee of the International Telephone and Telegraph (CCITT) are defined the image compression standards for both binary and continuous tone (monochrome and Cooler) images. Some of the Image Compression Standards are:1) JBIG1, 2) JBIG2, 3) JPEG-LS, 4) DCT based JPEG and 5) Wavelet based JPEG2000 Currently, JPEG2000 [3] is widely used because; the JPEG-2000 standard supports lossy and lossless compression of single-component (e.g., grayscale) and multi component (e.g., color) imagery. In addition to this basic compression functionality, however, numerous other features are provided, including: 1) progressive recovery of an image by fidelity or resolution; 2) region of interest coding, whereby different parts of an image can be coded with differing fidelity; 3) random access to particular regions of an image without the needed to decode the entire code stream; 4) a flexible file format with provisions for specifying opacity information and image sequences; and 5) good error resilience. Due to its excellent coding performance and many attractive features, JPEG 2000 has a very large potential application base. Some possible application areas include: image archiving, Internet, web browsing, document imaging, digital photography, medical imaging, remote sensing, and desktop publishing.

The main advantage of JPEG2000 over other standards, First, it would addresses a number of weaknesses in the existing JPEG standard. Second, it would provide a number of new features not available in the JPEG standard.

III. IMPLEMENTATION

Introduction of LSB: Data hiding is a method of hiding secret messages into a cover-media such that an unintended observer will not be aware selected as the cover media. These images are called cover of the existence of the hidden messages. In this paper, 8-bit grayscale images are cover-images with the secret messages embedded in them are called stego-images. Image quality is the measure of data hiding. One of the common techniques is based on manipulating the least-signi7cant-bit (LSB) planes by directly replacing the LSBs of the cover-image with the message bits. LSB methods typically achieve high capacity.

Simple LSB substitution: In this section, the general operations of data hiding by simple LSB substitution method are described.

Let C be the original 8-bit grayscale cover-image of $M_C \times N_C$ pixels represented as $C = \{x_{ij}\}$

$$C = \left\{x_{ij} \middle| 0 \le i < M_c, 0 \le j < N_c, x_{ij} \epsilon \{0, 1, .., 255\}\right\}$$

M be the n-bit secret message represented as

$$M = \{m_i | 0 \le i < n, m_i \epsilon \{0, 1\}\}$$

Suppose that the n-bit secret message M is to be embedded into the k-rightmost LSBs of the cover-image C. Firstly, the secret message M is rearranged to form a conceptually k-bit virtual image M_ represented as

$M' = \{m'_i | 0 \le i < n', m'_i \epsilon \{0, 1, \ldots\ldots, 2^k - 1\}\}$ Where $n' < M_c \times N_c$. the mapping between the n-bit secret message $M = \{m_i\}$ and the embedded message $M' = \{m'_i\}$

$$m'_i = \sum_{j=0}^{k-1} m_{i\times k+i} \times 2^{k-1-j}$$

Secondly, a subset of n_ pixels $\{x_{l1}; x_{l2}; \ldots; x_{ln}\}$ is chosen from the cover-image C in a prede7ned sequence. The embedding process is completed by replacing the k LSBs of x_{li} by m_i . Mathematically, the pixel value xli of the chosen pixel for storing the k-bit message m_i is modi7ed to form the stego-pixel x_li as follows:

$$x'_{li} = x_{li} - x_{li} mod 2^k + m'_i$$

In the extraction process, given the stego-image S, the embedded messages can be readily extracted without referring to the original cover-image. Using the same sequence as in the embedding process, the set of pixels $\{x'_{l1}, x'_{l2}, \ldots\ldots, x'_{ln'}\}$ storing the secret message bits are selected from the stego-image. The k LSBs of the selected pixels are extracted and lined up to reconstruct the secret message bits. Mathematically, the embedded message bits Mi' can be recovered by

$$m'_i = x'_{li} mod 2^k$$

PSNR of the obtained stego-image can be computed by

$$PSNR_{worst} = 10 \times log_{10} \frac{255^2}{WMSE}$$

$$= 10 \times log_{10} \frac{255^2}{(2^k - 1)^2} dB$$

Table 1 tabulates the worst PSNR for some k = 1–5. It could be seen that the image quality of the stego-image is degraded drastically when k>=4.

Table.1 Trade-off between speed and quality for kadak test set

	JPEG 2000 5/3			PGF		
Ratio	enc	dec	PSNR	enc	dec	PSNR
2.7	1.86	1.35	64.07	0.34	0.27	51.10
4.8	1.75	1.14	47.08	0.27	0.21	44.95
8.3	1.68	1.02	41.98	0.22	0.18	40.39
10.7	1.68	0.98	39.95	0.14	0.13	38.73
18.7	1.61	0.92	36.05	0.12	0.11	35.18
35.1	1.57	0.85	28.86	0.10	0.09	31.67

Optimal pixel adjustment process: To enhance the image quality of the stego-image obtained by the simple LSB substitution method, optimal pixel adjustment process (OPAP) is proposed. The basic concept of the OPAP is based on the technique proposed

Letp_i, p'_i and p''_i be the corresponding pixel values of the ith pixel in the cover-image C, the stego-image C' obtained by the simple LSB substitution method and the refined stego-image obtained after the OPAP. *Let* $\delta_i = p'_i - p_i$ be the embedding error between pi and pi' . According to the embedding process of the simple LSB substitution method, pi' is obtained by the direct replacement of the k least signi7cant bits of pi with k message bits, therefore$-2^k < \delta_i < 2^k$.The value of &i can be further segmented into three intervals, such that

$$Interval\ 1: -2^k < \delta_i < 2^k,$$

$$Interval\ 2: -2^{k-1} \le \delta_i \le 2^{k-1},$$

$$Interval\ 3: -2^k < \delta_i < 2^{k-1}.$$

Based on the three intervals, the OPAP, which modifies p'_i to form the stego-pixel p''_i, can be described as follows:

$$Case\ 1(2^{k-1} < \delta_i < 2^k) : If\ p'_i \ge 2^k, then\ p''_i = p'_i - 2^k;\ otherwise\ p''_i = p'_i;$$

$$Case\ 2(-2^{k-1} \le \delta_i \le 2^{k-1}) : p''_i = p'_i;$$

$$Case\ 3(-2^k < \delta_i < 2^{k-1}) : If\ p'_i < 256 - 2^k, \quad then\ p''_i = p'_i + 2^k;\ otherwise\ p''_i = p'_i.$$

$Let\ \delta'_i = p''_i - p_i$ be the embedding error between p_i and p''_i. δ'_i can be computed as follows:

$$Case\ 1\left(2^{k-1} < \delta_i < 2^k\ and\ p'_i \geq 2^k\right)$$

$$\delta'_i = p''_i - p_i = p'_i - 2^k - p_i = \delta_i - 2^k$$

$$\Longrightarrow 2^{k-1} - 2^k < \delta'_i < 2^k - 2^k$$

$$\Longrightarrow -2^{k-1} < \delta'_i < 0.$$

$$Case\ 2\left(2^{k-1} < \delta_i < 2^k\ and\ p'_i < 2^k\right)$$

$$\delta'_i = p''_i - p_i = p'_i - p_i = \delta_i$$

$$\Longrightarrow 2^{k-1} < \delta'_i < 2^k.$$

$$Case\ 3\left(-2^{k-1} \leq \delta_i \leq 2^{k-1}\right)$$

$$\delta'_i = p''_i - p_i = p'_i - p_i = \delta_i$$

$$\Longrightarrow -2^{k-1} < \delta'_i < 2^{k-1}.$$

$$Case\ 4\left(-2^k < \delta_i < 2^{k-1}\ and\ p'_i < 256 - 2^k\right)$$

$$\delta'_i = p''_i - p_i = p'_i + 2^k - p_i = \delta_i + 2^k$$

$$\Longrightarrow -2^k + 2^k < \delta'_i < -2^{k-1} + 2^k.$$

$$\Longrightarrow 0 < \delta'_i < 2^{k-1}.$$

$$Case\ 5\left(-2^k < \delta_i < 2^{k-1}\ and\ p'_i \geq 256 - 2^k\right)$$

$$\delta'_i = p''_i - p_i = p'_i - p_i = \delta_i$$

$$\Longrightarrow -2^k < \delta'_i < 2^{k-1}.$$

The 7rst set of cover-images consists of four standard grayscale images, ‘Lena’, ‘Baboon’, ‘Jet’ and ‘Scene’, each of 512 × 512 pixels, as depicted in Fig. 1. The second set consists of 1000 randomly generated grayscale images. Secret images are of two sets. The first set of secret message consists of 1000 randomly generated message of 512 × 512 × k bits, where k refers to the number of LSBs in the cover image pixels that are used to hold the secret data bits. For example, suppose that the last two LSBs of the cover image pixels are used to hold the secret data, then the secret data is of size 512×512×2=524 288 bits. The second set consists of the reduced-sized images of the grayscale image ‘TiL’ as shown in Fig. 1. The reduced-sized images are of size 512 × 256 pixels (for 4-bit insertion), 384×256 pixels (for 3-bit insertion), 256 × 256 pixels (for 2-bit insertion) and 256 × 128 pixels (for 1-bit insertion), respectively.

The results of embedding ssthe first set of secret messages into the first set of cover-images. Method with the optimal pixel adjustment process; the column labeled LSB is the simple LSB substitution method; and the column labeled OLSB in the optimal LSB substitution method proposed. The obtained PSNR values are the average values of embedding the 1000 sets random messages into the cover-images. For the OLSB method, for k = 1; 2, the obtained PSNR values are the average values of embedding the 1000 sets random messages into the cover-images, for k = 3, the obtained PSNR values are the average values of embedding the 10 out of 1000 sets random messages into the cover-images while for k = 4, no experiments are conducted due to the large number of searching space for the optimal substitution matrix. The results shows that the proposed method has better performance compared with LSB and OLSB methods for k = 2–4.

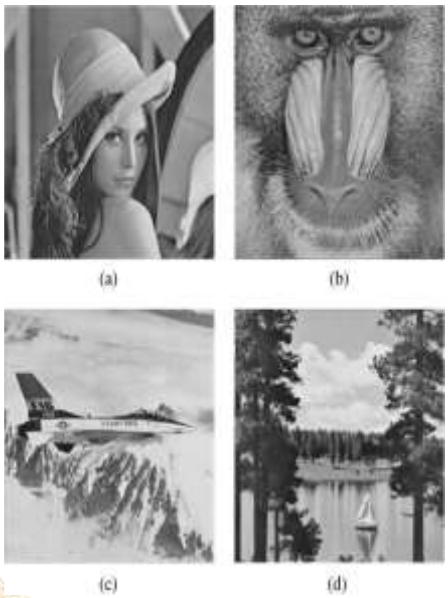

Fig.5. the first set cover-images of size 512 × 512 pixels.

The results of embedding the reduced-sized image of Fig. 1 into the first set of cover-images are listed in Table 3. The results also reveal that our proposed method has much better performance than the LSB and OLSB methods for k = 2–4.

Table 4 also shows the percentage of cover image pixels associated with the five cases:

$Case\ 1\left(2^{k-1} < \delta_i < 2^k \text{ and } p'_i \geq 2^k\right)$

$Case\ 2\left(2^{k-1} < \delta_i < 2^k \text{ and } p'_i < 2^k\right)$

$Case\ 3\left(-2^{k-1} \leq \delta_i \leq 2^{k-1}\right)$

$Case\ 4\left(-2^k < \delta_i < 2^{k-1} \text{ and } p'_i < 256 - 2^k\right)$

$Case\ 5\left(-2^k < \delta_i < 2^{k-1} \text{ and } p'_i \geq 256 - 2^k\right)$

With the rapid development of network technology, vast multimedia data would be communicated over the network. Although network transmission is convenient and fast, the multimedia data passing through the network is often attacked and tampered by malicious attackers. From the literatures many people are devoted to study the security for multimedia data.

Simulations and discussions, the EMD-scheme can enhance the capacity of secret message and the quality of the stego-image. Recently, Lee et al. proposed an improved data-hiding scheme, say LWC-scheme, which catches both of two adjacent pixels at a time and improves the possible situations from five to eight. As a result of LWC-scheme, it can promote the capacity 1.5 times approximately the former. Since the data embedding process uses the fixed evaluating parameters in both of EMD-scheme and LWC-scheme, they will be cracked easily and leak the secret message within the stego-image while their technology are disclosed.

Therefore, some concerns about the security issues will be considered. Later, Kuo et al. (for short KWSK-scheme) proposed two high capacity EMD data hiding techniques with changing-evaluating-value to improve the shortcoming of above schemes, in other words, the stego-images will still be safe even when it publishes the embedding formulas. According to KWSK-scheme, they used the synchronous generator of random numbers to minimize the possibility of message disclosure and improve the lack of open method but there is an open problem of synchronization of random seeds before the stego-image is transmitted between the sender and the receiver.

Least-Significant-Bit (LSB) Matching Method: In order to keep the embedding of the same amount of information as LSB matching and detect the secret data harder than the conventional LSB matching method, Mielikainen proposed a robust LSB matching method in 2006. There are two major properties in his scheme as following:

$$f(l-1,n) \neq f(l+1,n), \forall l,n \epsilon Z.$$

$$f(l,n) \neq f(l,n+1), \forall l,n \epsilon Z.$$

Therefore, embedding message is performed for two pixels *X* and *Y* of a cover image at a time and then adjusting one pixel of the (*X*, *Y*) to embed two secret bits message *s1s2*. The embedding flowchart is shown in Fig.2 and the embedding procedure is described as following:

Step.1. If the LSB of *X* is the same as *s1*, go to step 2. Otherwise, go to step 3.

Step.2. If the value of $f(X, Y)$ is the same as *s2*, do not change any pixel. Otherwise, the value of pixel *Y* is increased or decreased by 1.

Step.3. If the value of $f(X-1, Y)$ *is* the same as *s2*, the value of pixel *X* is decreased by 1. Otherwise, the value of pixel *X* is increased by 1. Where the function $f(X, Y)$ is defined as Eq.1:

$$f(X', Y') = LSB\left(\left\lfloor \frac{X'}{2} \right\rfloor + Y'\right)$$

Since this new LSB matching method just only increase or decrease 1 in two adjacent pixels, the difference of the two neighborhood pixel between cover image and stego-image is very small. Hence, it can keep high quality while hiding data.

Hiding Secret Files is a security product based on an unrivalled data hiding method. As a result of research, this product hides and protects your private information from being erased. You would not be happy to find your financial plans, personal ideas or projects, your private photos or movies are altered or erased by foreign hands. Hiding Secret Files is the ultimate solution to have exclusive secured access to sensitive information based on a password. No one except you will be able to access the secured data. It cannot be changed even with own operating system. The latest, and one of the most annoying secondary products of the Internet evolution, is the spy-ware activity, and thankfully it is absolutely inoffensive against the protection guaranteed by Hide Secret Files. The hidden data won't be visible even for your own OS so the Safe Mode booting or moving the hard drive in another PC won't make it possible to reveal, the data protected by Hide Secret Files. WinRAR is to be installed on your computer so that it can work and Microsoft Windows with access to the command prompt. If WinRAR is not installed on your computer this can be downloaded through our recommended download section.

Hiding a message or other data:

1. Create a text file with your secret message or hidden data and highlight it and highlight each of the files you wish to secretly add to the image file. In this example we created one text file called**message.txt**.
2. Once highlighted right-click the highlighted file and click **add to "message.rar"**, where message.rar is the name of the file you right-clicked on.
3. Open a Windows command line window.
4. Move to the directory that contains the .rar file and the image you wish to hide the text in.
5. Type a command similar to the below command.

   copy /b secret.jpg + message.rar hidden.jpg

In the above example, "secret.jpg" is the name of the image you're using, the .rar file is the name of the file used earlier, and hidden.jpg is the name of the new image with the hidden message within it. See the copy command page for additional information about this command.

Once the above steps have been completed you should now have an image called hidden.jpg that contains the hidden message. It's a good idea to make sure you're still able to open and view the image before saving it, posting it on the Internet, or otherwise distributing it. Below is an example of the hidden.jpg we created doing the above steps.

To view the hidden message or hidden files you must have followed the above steps. If the above steps were performed to create the image follow the below steps to view the data.

1. Save the image to the computer if you're viewing it online.
2. Open WinRAR by clicking Start, Programs, WinRAR, and then WinRAR.
3. Within WinRAR click File and Open archive. Within the open window make sure your files of type option are all files and not just compressed files.
4. Browse to the location of the image and double-click the image to open it.
5. Once open it should display the file(s) contained within the image that can be extracted from the image.
6. How to use data selection to explore available data and drill down to selected properties
7. Using the data comparison condition
8. Using the set a data value action
9. That Rules recognizes different *types* of data, and verifies when necessary
10. That Rules knows that not all data is writable, and verifies when necessary
11. How to create composite tokens, extending the tokens listed in the replacement patterns
12. Making field values accessible to Rules

13. Using reference fields to access new data, such as tags on an article or nodes in a node reference field

Steganography replaces the bits which are not needed in image and sound files with secret data. Instead of data protecting in encryption, steganography hides the every existence of the data. It is also undetectable under traditional traffic-pattern analysis. There are few lawful uses for steganography, e.g., forensics professionals and reports circulating about terrorists using steganography to communicate secretly, experts doubt. "Most people study steganography either as an academic discipline or a curiosity, but I don't know if even terrorist groups would actually use it," says Chakraborty. After reading a USA Today article about steganography and terrorism, Neils Provos, a Ph.D. student in computer science at the University of Michigan in Ann Arbor, decided do his dissertation on steganography. Provos developed detection and cracking tools to analyze images for signs of steganography, such as overly large files and uneven bit mapping. He tested the tools and then used them to compare 2 million images on San Jose-based eBay Inc.'s Web site, which has been cited as a possible place for posting and retrieving hidden messages. "Steganography becomes the focus of attention, dies down, and then the public is all over it again," says Provos. "But it will never be pervasive, because the amount of data you can actually hide in the images is fairly small. And if someone wanted to steal intellectual property, it'd be easier to copy the data on a disk and carry it out in your pocket." Even if steganography is present, forensics experts prefer to start by investigating less complex areas. But in some cases, the only evidence might be hidden in image or sound files, so investigators need to be aware of steganography and the tools used to detect and crack it.

IV. RESULTS

Using MATLAB visual basics develop the explorer windows and observed output result

Input selection processes:

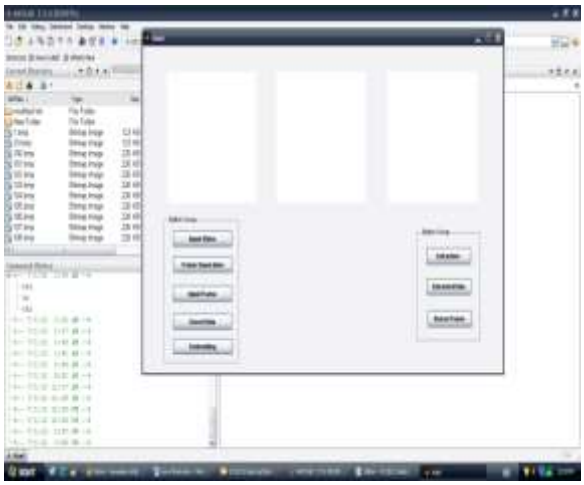

Fig.6

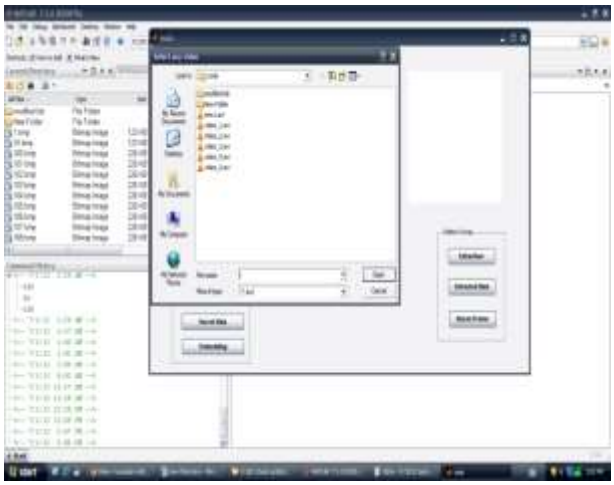

Fig.7

Frame selection processes:

Selecting frame:

Fig.8

Select secrete data processes:

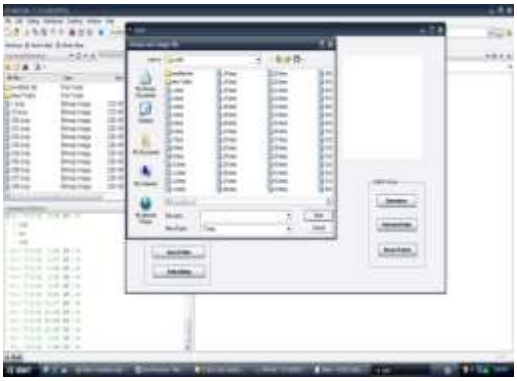

Fig.9

Extracting processes:

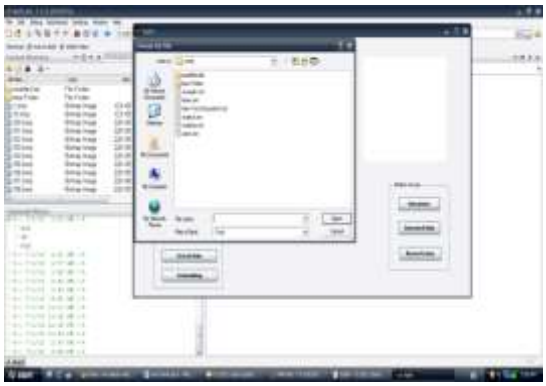

Fig.10

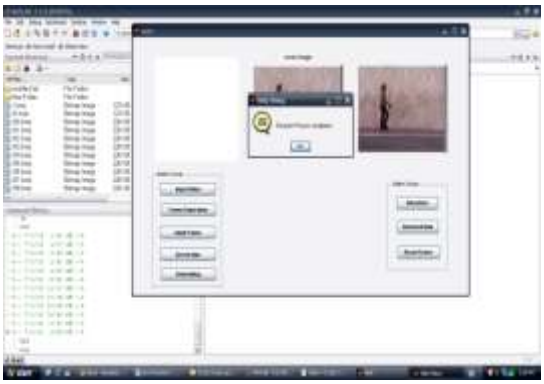

Fig.11

Extracting data:

Fig.12

V. CONCLUSION

In this paper, a data hiding method by simple LSB substitution with an optimal pixel adjustment process is proposed. The image quality of the stego-image can be greatly improved with low extra computational complexity. Extensive experiments show the effectiveness of the proposed method. The results obtained also show significant improvement than the method proposed with respect

to image quality and computational efficiency. Steganography transmits secrets through apparently innocuous covers in an effort to conceal the existence of a secret. Digital image steganography and its derivatives are growing in use and application. In areas where cryptography and strong encryption are being outlawed, citizens are looking at steganography to circumvent such policies and pass messages covertly. As with the other great innovations of the digital age: the battle between cryptographers and cryptanalysis, security experts and hackers, record companies and pirates, steganography and Steganalysis will continually develop new techniques to counter each other.

In the near future, the most important use of steganographic techniques will probably be lying in the field of digital watermarking. Content providers are eager to protect their copyrighted works against illegal distribution and digital watermarks provide a way of tracking the owners of these materials. Steganography might also become limited under laws, since governments already claimed that criminals use these techniques to communicate.

REFERENCES

[1] A.Z. Tirkel, R.G. Van Schyndel, C.F. Osborne, A digital watermark, Proceedings of ICIP 1994, Austin Convention Center, Austin, Texas, Vol. II, 1994, pp. 86–90.

[2] W. Bender, N. Morimoto, A. Lu, Techniques for data hiding, IBM Syst. J. 35 (3/4) (1996) 313–336.

[3] T.S. Chen, C.C. Chang, M.S. Hwang, A virtual image cryptosystem based upon vector quantization, IEEE Trans. Image Process. 7 (10) (1998) 1485–1488.

[4] L.M. Marvel, C.G. Boncelet, C.T. Retter, Spread spectrum image steganography, IEEE Trans.Image Process. 8 (8) (1999) 1075–1083.

[5] Anderson R.J. and Petitcolas F.A.P., "On the Limits of steganography," J. Selected Areas in Comm., vol. 16, no.4,1998, pp. 474–481.

[6] Bailey, K. and Curran, K. "An evaluation of image-based steganography methods". International Journal of Digital Evidence, Fall 2003.

[7] Chapman, M. Davida G, and Rennhard M.. "A Practical and Effective Approach to Large-ScaleAutomatedLinguistic Steganography" found online at http://www.nicetext.com/doc/isc01.pdf

[8] Dai Y., Liu G., and WangBreaking Z., "Predictive - Coding-Based Steganography and Modification for Enhanced Security", IJCSNS International Journal of Computer Science and Network Security, vol.6 no. 3b, March 2006