Safer data transmission using Steganography

Arul Bharathi, B.K.Akshay, M.Priya, K.Latha

Department of Computer Science and Engineering Sri Sairam Engineering College
Chennai, India
Email: arul.bharathi@yahoo.com, akshaybarade@gmail.com,
mpriya092@gmail.com, lathu_2k@yahoo.com

***Abstract*- Today's Internet connected networks are under permanent attack by intruders and hackers. The explosive growth of computer systems and their interconnections has led to a heightened awareness of the need to protect the data transmitted. There are many ways already existing and also discovered daily. The field of cryptography has got more attention nowadays. There exist many encryption algorithms like RSA, DES, and AES etc. But still hackers are able to retrieve the messages which are sent secretly. So, to deceive the hackers, people have started to follow a technique called 'Steganography'. In this method, the data is hidden behind unsuspecting objects like images, audio, video etc. so that people cannot even recognize that there is a second message behind the object. Images are commonly used in this technique. In this paper, we have proposed an idea for overcoming some serious drawbacks in steganography and we have tried to overcome those with our ideas. Here, the pixels in the images are replaced with the new ones, which are almost identical to the old ones, in a manner that can be used to retrieve back the hidden data.**

1.INTRODUCTION

Steganography is the art and science of communicating in a way which hides the existence of the communication. In contrast to cryptography, where the "enemy" is allowed to detect, intercept and modify messages without being able to violate certain security premises guaranteed by a cryptosystem, **" *the goal of steganography is to hide messages inside other harmless messages in a way that does not allow any enemy to even detect that there is a second secret message present."***

2. STEGANOGRAPHY IN IMAGES

In steganographic images any plain text, cipher text, other images, or anything that can be embedded in a bit stream can be hidden in an image. The common methods followed for hiding data in images are the **'Least Significant Bit (LSB) Insertion'** technique in which the LSB of the pixel values are replaced with the data to be encoded in binary form, the '**Masking Technique'** in which the original bits are masked with data bits and the **'Filtering Technique'** in which certain transformations are done on the image to hide data. This follows the same procedure as that of water marker.

3. DRAWBACK IN THE CURRENT TECHNIQUES

When the pixels will be scanned for a possible relation it will be used to trace out the actual characters. Only 24 bit messages are suitable and 8 bit images are to be used at great risk. Extreme Care needs to be taken in the selection of the cover image, so that the changes to the data will not be visible in the stego-image.

4. THE PIXEL REPLACEMENT TECHNIQUE

To overcome the drawbacks in the currently followed techniques, we propose a new methodology for hiding data in images. Here, *we place the existing pixels in the image with the new ones in such a way that no difference is visible between the contaminated and the original image.*

5. PROCEDURE TO IMPLEMENT THE TECHNIQUE

A. Encoding

The Algorithm used for encoding in this technique can be described with the following steps:

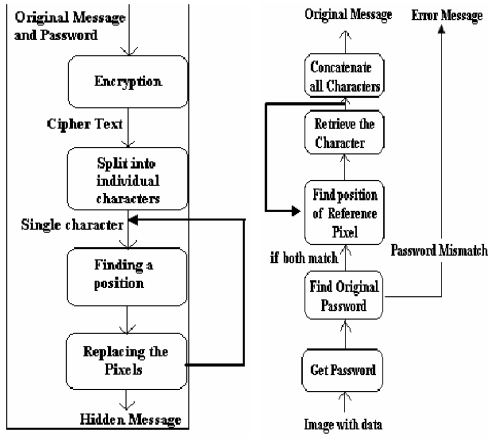
- *Get the Image, Message to be hidden and the Password.*
- *Encrypt the message and the password.*
- *Move some rows below the first row in the image and fix a reference position*
- *Near the left edge for odd characters and near the right for even characters.*
- *For each character in the original data do*
- *find a position corresponding to that character*
- *search the surrounding pixels and find a pixel value closer to all of them*
- *replace the current pixel and the reference pixel with this value*
- *move to the next row*
- *Set a pixel value as a threshold.*
- *Repeat steps 4 and 5 for the Password from the bottom of the image.*

B. Decoding

The Algorithm for retrieving the original message from the steganographed image follows this sequence:

- *Get the Image and the Password.*
- *Move to the bottom row in the image where password hiding starts.*
- *Find the value of the reference pixel.*
- *Search the entire row for the same pixel value.*
- *Find the position of that pixel and decode the character.*
- *Repeat steps 3 thro 5 till the threshold is reached.*
- *Concatenate all the characters found so far (Actual Password).*
- *If the found password does not match the given password go to step 11.*
- *Move to the top row in which the first character of original data was stored.*
- *Repeat the sequence followed in steps 3 to 7 to get the original message.*
- *Display the result.*

6. FLOW DIAGRAM



(a) (b)

Fig. 1: (a) encoding (b) decoding

7. IMPLEMENTATION

With this algorithm described, let us describe the pixel replacement technique in detail. First, we shall see how the original message and the password are hidden into the image and then we'll discuss how to retrieve message for the authorized person who knows the correct password.

A. Encrypting the Message and providing Password

The actual message and the password are obtained from the user. Then, they are encrypted using any of the encryption algorithms like **RSA, DES** etc. This encryption step provides an additional safety feature added to the technique to ensure maximum safety.

B. Choosing a position to hide the character

The actual process of steganography starts begins here. The Image is scanned from the top row – wise. Few rows are omitted. The message to be sent is split into individual characters. The following process is repeated for all the characters in the message. A position for hiding the character is chosen according to **some relation with that character**. The relation can be something like the ASCII value of the character, the order of occurrences of that character in the Alphabetical or Reverse order if it is an alphabet etc.

For example, the position of the character 'R' can be chosen as:
ASCII Value of 'R' = **82**,
So, position = 82 – 50 = **32.**
(only e.g. It need not be 50 to use)

C. Finding a Suitable Color

Once a position is chosen, the values of all the pixels surrounding the pixel in that position are found. Since this position is usually not near the edge of the image, there will be 8 pixels surrounding it. A pixel value, i.e., **a color**, is chosen so that it **does not differ much** from those of the 8 pixels. This is the most difficult step in the whole process. The value will differ only by a small value. Such a small change in the color will be indiscernible to the users.

D. Replacing the pixels

Now, with the color to be replaced being found, it's time to replace the pixels with the new color. A position near to the left side of the row is fixed as the reference position for the odd numbered character i.e., the 1st, 3rd, 5th character and so on. Similarly, for the even numbered characters, a position close to the right side of the row is selected as the reference position. These reference positions are the same for all the characters. When the reference position is chosen, **the pixel in that position is replaced with the new color.** Then, the pixel in the already found position for the character (in our e.g., this is 32) is also replaced with the new color.

Example of contaminated image

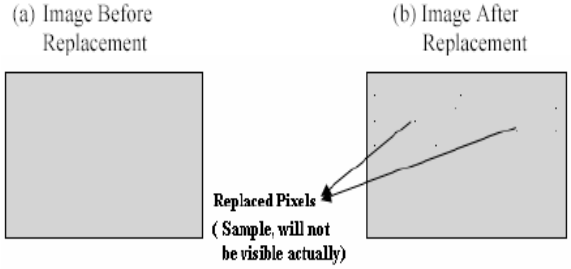


Fig. 2: contaminated image

Here is such an example where the image is contaminated with data to be sent. But, in real situations, these will not be visible at all because the color chosen for replacing is so close to the original color that cannot be found by the human eye but still can be found out by the computer. The above mentioned steps of finding a suitable color and replacing the pixels is continued for all the characters in the encrypted message.

E. Hiding the Password

The same steps of choosing the color, replacing the pixels and setting a threshold are repeated for hiding the password but the only difference is that the image is scanned row – wise **from the last row** instead of from the first row.

F. Setting a Threshold

The final step is to set a threshold pixel in a fixed position **to indicate the end** of the encrypted message. This is essential for decoding the message from the image. Otherwise, we cannot find the end of the message.

Some methods to determine Variable Threshold:

For determining variable threshold is an important job. Various ways are suggested for finding a proper threshold for an image. Determining a proper value largely depends on a close and precise study of the nature and quality of the image. The color-concentration, number of edges and contrast of colors are some of such properties. Two methods which give satisfactory threshold value for images. The first method does a quantitative study of the number of edges present in the image. The second plots a frequency graph for all the color components and thus, determines an approximate concentration of colors.

a) A picture with a lot of edges and random pixels
b) A uniform color image

Fig.3: Edge finding method

This procedure is the inference that larger the number of edges in an image, higher the number of adjacent pixels with highly contrasting colors.

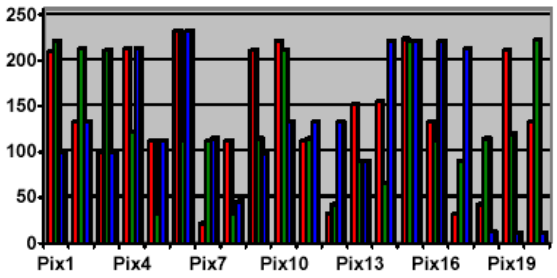


Fig.4: Frequency plotting of an image with a sharp slope of RED component

This method scans the image and records the frequency of occurrence of all the combinations of each of the RGB components.

Retrieving the Message

The process of retrieving the original message from the steganographed image is similar to that of the hiding process except **in the reverse order**. First, the password is got from the user. The image is scanned from the last row and the row in which the password hiding started is reached. The reference pixel value is found and the position of that color in that row is noted. Then, again the relation used previously for finding the position is used to get the original character. This can be explained as:
If the position is **32**, then
32 + 50 = **82**,
The character of ASCII value 82 is **'R'.**

Similarly, the other characters are found till the threshold is reached and all of them are concatenated to get the **original password**. Now, the given password and the original are checked and if they match, then further processes are done, otherwise, an **error message** is displayed.

If the password matches, then the image is scanned from the top and the starting
Row from where the data hiding started is reached. The same steps of finding the reference pixel and the position of the other pixel are repeated again till the threshold.
Then, all the characters are joined and the original message is displayed.

As seen from this example, there are no big changes visible between the actual image and the one in which data is hidden. Thus, our proposed technique can be a useful method for hiding messages in images.

Fig.5: Illustration

8. PRESENT DAY SCENARIO AND WORLD SURVEY

Nowadays, Steganography is not the same as cryptography. Data hiding techniques have been widely used to transmission of hiding secret message for long time. Ensuring data security is a big challenge for computer users. Business men, professionals, and home users all have some important data that they want to secure from others. Even though both methods provide security, to add multiple layers of security it is always a good practice to use Cryptography and Steganography together. By combining, the data encryption can be done by a software and then embed the cipher text in an image or any other media with the help of stego key. The combination of these two methods will enhance the security of the data embedded. This combined chemistry will satisfy the requirements such as capacity, security and robustness for secure data transmission over an open channel. The resulting stego-image can be transmitted without revealing that secret information is being exchanged. Furthermore, even if an attacker were to defeat the steganographic technique to detect the message from the stego-object, he would still require the cryptographic decoding key to decipher the encrypted message. Since then, the steganography approaches can be divided into three types:
Pure Steganography: This technique simply uses the steganography approach only without combining other methods. It is working on hiding information within cover carrier.
Secret Key steganography: The secret key steganography use the combination of the secret key cryptography technique and the steganography approach. The idea of this type is to encrypt the secret message or data by secret key approach and to hide the encrypted data within cover carrier.

Public Key Steganography: The last type of steganography is to combine the public key cryptography approach and the steganography approach. The idea ofthis type is to encrypt the secret data using the public key approach and then hide the encrypted data within cover carrier. But what we are introducing and proposing is the new technique where pixel level implementation is taken into consideration. Thus by the overall survey around the world , the exsiting methods and process of steganagraphy can be made efficient by this method.

9. ADVANTAGES OF OUR TECHNIQUE

- Cost of cracking the hidden message is extremely high.
- The data cannot be easily decoded without the key using Image manipulation technique.
- Any type of image, 8 or 24 bits can be used.
- There is no increase in the size of the image due to data in it.
- There are no constraints on the choice of the image.

10. CONCLUSION

To overcome the drawbacks in the existing cryptography and steganography techniques, we have proposed a new technique for hiding data in images. Our technique is less prone to attacks and since the data is strongly encrypted and the cost of retrieving it by unauthorized persons is extremely high. Since the pixels are replaced with almost identical pixels, it is difficult to even identify that there is a second message hidden. So, we hope that our technique will be used widely in the future.

10. REFERENCES

[1] William Stallings, 3rd Edition Cryptography and Network Security/principles and practices.
[2] http://ijrte.academypublisher.com/vol01/no01 /ijrte0101672674.pdf
[3] http://www.scribd.com/doc/28171077/Pixel-Replacement-Technique-Using-Steganography-A-High
[4]http://airccse.org/journal/sipij/papers/1210 sipij06.pdf
[5] Nader F.Mir,"Computer and Communication Networks", Pearson Education, 2007
[6]http://diit.sourceforge.net/files/HidingBehindCorners.pdf
[7]http://martinolivier.com/open/stegoverview.pdf