**A COMPARISON STUDY BETWEEN DIFFERENT TYPES OF STEGANOGRAPHY METHODS**

**ABSTRACT**

There are a wide variety of steganography methods that are running wild nowadays. Steganography is hiding image inside an image file, audio file etc.. This paper compares some of the methods that are used in steganography both covered and coverless in terms of security, robustness, merits and demerits. While some of them focuses on security and robustness some other give more importance to undetectability and so. Various techniques from various papers are taken for this study which are referenced below.

**INTRODUCTION**

Steganography is the technique of hiding secret data within an ordinary, non-secret, file or message in order to avoid detection; the secret data is then extracted at its destination. The use of steganography can be combined with encryption as an extra step for hiding data. The word steganography is derived from the Greek words steganos (meaning hidden or covered) and the Greek root graph (meaning to write).

Steganography can be covered or coverless. Covered steganography is the ordinary steganography itself where the image pixels are manipulated or changed in order to hide the information. But in coverless image we do not need a designated cover image, the data will be hidden in plain sight or without modifying the picture itself.

**COVERLESS IMAGE STEGANOGRAPHY WITHOUT EMBEDDING**

Main idea

This paper finds the images whose hash functions are the same as the secret data.

Proposed method

* An image database is collected.
* Hash sequence generated by a hashing algorithm that is share between sender and receiver.
* Secret data converted to bit stream and divided into sequence of segments.
* For each segment the image whose hash function is the same as the segment is found using an inverted index structure.
* These series of images associated with the segments are sent.
* Hashing algorithm: covert image into 3x3 matrix; compute average intensity of each matrix; concatenate them in zig-zag manner to get I1,I2…In;

Hash sequence generation = { hi =1 if Ii>=Ii+1

Hi =0 otherwise

Comparison:

This method is compared with the usual covered steganography techniques LSB replacement and LSB matching using the SER values and it is seen that it is robust to attacks such as rescaling, luminance, contrast in which it had zero as SER value and for JPEG compression it was far lesser than the others.

The advantages that this method had in addition to robustness to various typical image attacks are resistance to steganalysis resistance and no designated cover image. But on the other side it had a limited payload, only 8 bit data can be added in each image and if the hash algorithm is leaked or not robust enough the whole method collapses.

**MODIFIED HILL CIPHER ALGORITHM USING MYSZKOWSKI TRANSPOSITION TO ADDRESS KNOWN PLAIN TEXT**

Main idea:

The paper proposes a method in which the Hill cipher algorithm is combined with layers of Myszkowski transposition method to increase the security of the message sent.

Proposed method:

The method had 3 phases :

* Generation of key - Since both Hill cipher and Myszkowski need keys, they are generated using an algorithm through a 3x3 matrix; at the end of the key generation phase we will be having 3 keys(k1,k2,k3) each consisting of a sequence of numbers.
* Encryption phase – Arrange the input message into rows of same length with the key. Let that be denoted as P. Apply Myszkowski technique using k2, to that output apply Myszkowski technique using k3, to that output apply Hill cipher using K1 and at last to that resultant output apply a third round of Myszkowski algorithm using k4(any of the 3 keys). The output will be the encrypted data to sent.
* Decryption – The decryption is the exact reverse order of the encryptions using the keys in reverse order.

Comparison:

The performance and security of this proposed method is measured using the parameters such as known plaintext attack, randomness test, runtime test and avalanche effect. Since there no linear relationship between the plaintext and cipher text here and because the plaintext and ciphertext are put through multiple rounds of the incorporated transposition technique, the known plaintext attacks becomes more difficult. All the randomness tests passed with a p value>.01. Also the even though there are multiple layers of encryption the runtime taken was moderately medium. Runtime Test in Milliseconds demonstrates a 25.70% faster encryption time based on previous studies.

**COVERLESS INFORMATION HIDING BASED ON ROBUST IMAGE HIDING**

Main idea:

A new coverless steganography method based on robust image hashing.

Proposed method

* SIFT features are used to generate an image hash.
* Image content is not modified by this method, here based on SFIT features they are generating a hash value of 18-bits for an image.
* Using set of such images they are creating a database.
* The secret message is divided to segments of messages of 18-bits, if the last piece of message is short of 18-bits they are adding zeros to complete the 18-bits sequence.
* Here they are using SFIT features like sharp edges, corners, and brightness to find stable points, they divide an image to 3\*3 segments and checks for stable points, if the point is in particular region of 3\*3 segment they give a certain binary code like 00 or 01 as one want.
* Then each secret message segment is compared with image database and selects image, in such a way a series of images are selected in order and an extra image is selected that stores the length of the message using LSB replacement.
* The message is extracted based on the order of image block and length of image. For more integrity the data length of last image is calculated as s=l%18 if l%18! =0 or s= 18 if l%18==0.

**Comparison**

* Compared to “Coverless image steganography without embedding (paper 1)” this method has more capacity and robustness.
* In other method 8-bits are transmitted where as our method transmits 18-bits. This method uses only 46 images to transmit 100b where as other use 100, almost doubled the capacity
* Proposed method is robust to attacks like content damage, JPEG compression, rescaling etc., as we use SFIT.
* The main disadvantage is the large database needed to improve the capacity.