THE ART OF SECRET COMMUNICATION:EXPLORING STEGANOGRAPHY METHODS

1. **Mr.Panduraju Pagidimalla**
Assistant Professor,
Computer Science and engineering,
Mail:p.panduraju@gmail.com.

1. K. **YASHWANTH**
dept. Computer Science Engineering,
KL University,
Hyderabad, Telangana

2. G. **PAVAN REDDY**
dept. Computer Science Engineering,
KL University,
Hyderabad, Telangana

3. **VIPUL REDDY**
dept. Computer Science Engineering,
KL University,
Hyderabad, Telangana

4. L. **KEERTHANA REDDY**
dept. Computer Science Engineering,
KL University,
Hyderabad, Telangana

5. S. **PADMASHREE CHOWDARY**
dept. Computer Science Engineering,
KL University,
Hyderabad, Telangana

***Abstract*— The proposed algorithm uses binary codes and pixels within an image to hide data through steganography. The Steganography Imaging System (SIS) is developed to implement this algorithm, which is tested and found to be effective in hiding data of different sizes within the image.**

***Index Terms*— Steganography, Binary codes, Pixels, Zipped file, Steganography Imaging System (SIS),Image processing attacks, Data hiding.**

I. INTRODUCTION

The paper presents the Steganography Imaging System (SIS), an algorithm that hides data within images to improve data privacy. It discusses the importance of data security, compares steganography to other related technologies, and provides an overview of the different types of steganography, performance metrics, carrier file formats, and system implementation, and presents results from testing with varying data sizes.

II. PROCEDURE

A. IMAGE STEGANOGRAPHY

Image steganography is the practice of hiding information within an image to achieve covert communication. Various techniques can be used for image steganography, such as modifying the pixel values of the image or transforming the image using a Fourier transform. Steganography can be used for legitimate purposes, such as protecting confidential data or proving ownership or authenticity, but it can also be used for nefarious purposes, and sophisticated techniques can be challenging to detect.

III. METHODS

A. IMAGE FILES

Steganography is commonly used with image files to hide messages. The message is embedded in a way that is hard to detect by the naked eye, often in the least significant bits of the image pixels or the frequency domain of the image. Despite the potential image quality degradation caused by the process, image files remain popular due to their wide usage, easy shareability, and large data storage capacity.

B. STEGANOGRAPY ALGORITHM TO HIDE SECRET MESSAGE INSIDE AN IMAGE

- The system is designed to ensure the privacy, confidentiality, and accuracy of data.
- A framework for the system process is shown in Fig. 1, which includes data hiding and retrieval from images.
- Prior to data hiding, the user is required to log in with a username and password, and a secret key is needed to retrieve the hidden data.
- A novel steganography algorithm is used to embed data with minimal distortion to the original image.
- Fig. 2 shows the algorithm for embedding secret messages in the image, which requires a secret key for retrieval.
- The secret message is first transferred to a text file, compressed into a zip file for security, and converted into binary codes.
- Data hiding is then applied by encoding binary codes into pixels in the image.
- The secret key plays an essential role in the proposed steganography algorithm.

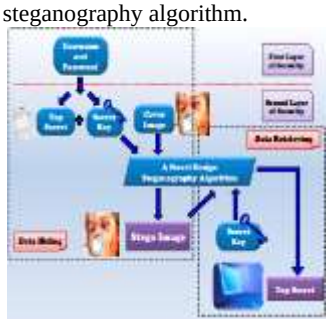

Fig:1 the frame work of the system

```
Begin Input: Cover_Image, Secret_Message, Secret_Key;
Transfer Secret_Message into Text_File; Zip Text_File;
Convert Zip_Text_File to Binary_Codes;
Convert Secret_Key into Binary_Codes;
Set BitsPerUnit to Zero;
Encode Message to Binary_Codes;
Add by 2 unit for bitsPerUnit;
Output: Stego _ image
End
```

Fig2: algorithm for embedding data inside image

- The steganography algorithm involves transferring the secret message into a text file, compressing it into a zip file, converting it into binary codes, and using data hiding to encode the binary codes into the pixels of the image.
- The secret key acts as a locker to lock and unlock the secret message, and each last two bits of the binary codes are encoded into each pixel of the image to minimize changes to the original image.
- To extract the secret message from the stego image, a secret key is needed for verification, and the binary codes are decoded to form a zipped text file, which is then unzipped to retrieve the original secret message.
- Fig. 3 presents the algorithm for extracting the hidden message from the stego image.
- A secret key is required to extract the message correctly from the image for verification purposes.
- The data extraction method shown in Fig. 3 also requires a secret key to check if it matches the key that decodes from the binary code sequence.
- Once the key is verified, the binary code is converted into a zipped text file.
- The text file is then unzipped, and the secret message is transferred from the file to retrieve the original message.

```
Begin Input: Stego_Image, Secret_Key;
Compare Secret_Key;
  Calculate BitsPerUnit;
  Decode All_Binary_Codes;
  Shift by 2 unit for bitsPerUnit;
  Convert Binary_Codes to Text_File;
  Unzip Text_File;
  Output Secret_Message;
End
```

Fig. 3 Algorithm for extracting data from stego image.

- The proposed steganography algorithm focuses on several key techniques, including transferring the secret message to a text file, zipping the file, converting the file and key into binary codes, and encoding those codes into the pixels of an image.
- Despite these techniques, the image quality remains robust with minimal distortion and color changes.
- The secret message is difficult to steal through steganalysis, which makes it a secure way to protect data from unauthorized access.
- The algorithm uses two image embedding techniques, namely data hiding and data retrieving methods.
- Data hiding is used to hide the secret message and key in the cover image, while data retrieval is used to extract the key and hidden secret message from the stego image.
- Overall, this steganography algorithm provides a secure way to protect data, especially secret messages, within an image without revealing them to unauthorized parties.

IV. FINDINGS

- The Steganography Imaging System (SIS) is a system that implements a proposed algorithm for hiding data inside an image. The system has two layers of security, with the first layer used for login and the second layer used for hiding and retrieving data as seen in the Fig[1].
- The main interface of the system consists of two boxes, one for the image and another for the data that the user wants to hide inside the image as shown in Fig[4]. The image box is used for getting the image from any location and the text box is used for hiding and retrieving the message to and from image respectively.

Fig. 4 The main interface for SIS.

- A secret key is required to ensure the security of the data, which is entered twice for verification purposes. The secret key, along with the hidden data, is embedded inside the image As shown in Fig[5]. The user can then save the new stego image to a different file. Overall, SIS provides a simple and efficient way to hide data inside an image with a high level of security.

Fig. 5 The secret key is required for SIS.

- This system allows users to hide secret data inside images, creating a new stego image that can be shared over the internet or email without revealing the hidden message. To retrieve the message, the recipient can upload the stego image and use the system to unlock the data using a secret key. The system was tested with images in Fig's. [6-7], where the stego images showed no noticeable distortion compared to the original images, even though their size was slightly larger.

Fig. 7 (a) Original image (b) Stego image

We then tested the algorithm using the PSNR (Peak signal-to-noise ratio). PSNR is a standard measurement used in steganography technique in order to test the quality of the stego images. The higher the value of PSNR, the more quality the stego image will have.

- If the cover image is C of size M × M and the stego image is S of size N × N, then each cover image C and stego image S will have pixel value (x, y) from 0 to M-1 and 0 to N-1 respectively.
- The PSNR is then calculated as follows:
  - **PSNR=20log10(MAX/(MSE)^(1/2)).**
- The PSNR block computes the peak signal-to-noise ratio, in decibels, between two images. This ratio is used as a quality measurement between the original and a compressed image. The higher the PSNR, the better the quality of the compressed, or reconstructed image.
- Note: MAX is the maximum possible pixel value of the images. For example, if the pixels are represented using 8 bits per sample, then the MAX value is 255. If the stego image has a higher PSNR value, then the stego image has more quality image

A. Steganography Algorithm to Hide Secret Message inside an Image.

- PSNR values show that stego images have quality without compromising original image
- Cover image must have minimum pixel requirements of width (150) and height (112) for data hiding
- Smaller BMP images (1.0 MB) can hide secret messages
- Proposed algorithm uses BMP format due to its simplicity and wide acceptance in Windows programs
- BMP files are uncompressed and have larger pixel size, providing more space for binary codes
- Zip technique used to reduce file size and enhance security
- Testing showed different sizes of data can be stored in BMP images using proposed algorithm (Table 2)
- Proposed algorithm can hide zipped file up to 6.93 KB in a 3.14 MB BMP image
- This can encode 27287 characters with spaces (or 4478 words or 10 pages) with near-zero distortion.

The proposed steganography algorithm is proven to be strong and robust, as it can produce a stego image without compromising the quality of the original image. The algorithm uses BMP image format, which is widely accepted in Windows programs and provides more space for binary codes to be encoded. To increase the amount of data that can be hidden, the algorithm uses zip technique to reduce the total size of the file and enhance its security. The biggest size of a zipped file that can be hidden in a 3.14 MB BMP image is 6.93 KB, which can encode 27287 characters with spaces (or 4478 words or equally to 10 pages of words) underneath the image with near-zero distortion.

V. CONCLUSION

This paper proposed a new steganography algorithm with 2 layers of security. A system named SIS (Steganography Imaging System) has been developed using the proposed algorithm. We tested few images with various sizes of data to be hidden. With the proposed algorithm, we found that the stego image does not have a noticeable distortion on it (as seen by the naked eyes). We also tested our stego images using PSNR value. Based on the PSNR value of each images, the stego image has a higher PSNR value. Hence this new steganography algorithm is very efficient to hide the data inside the image SIS can be used by various users who want to hide the data inside the image without revealing the data to other parties. SIS maintains privacy, confidentiality and accuracy of the data. Acknowledgments This research is supported under the Fundamental Research Grant Scheme (FRGS) Vot 0738.

VI. REFERENCES

[1] M. Chen, N. Memon, E.K. Wong, Data hiding in document images, in: H. Nemati (Ed.). Premier Reference Source– Information Security and Ethics: Concepts, Methodologies, Tools and Applications, New York: Information Science Reference, 2008, pp. 438-450.

[2] [2] D.C. Lou, J.L. Liu, H.K. Tso, Evolution of information – hiding technology, in H. Nemati (Ed.), Premier Reference Source–Information Security and Ethics: Concepts, Methodologies, Tools and Applications, New York: Information Science Reference, 2008, pp. 438-450.

[3] [3] Schneider, Secrets & Lies, Indiana:Wiley Publishing, 2000.

[4] [4] E. Cole, Hiding in Plain Sight: Steganography and the Art of Covert Communication, Indianapolis: Wiley Publishing, 2003.

[5] [7]Moerland, T., "Steganography and Steganalysis", Leiden Institute of Advanced Computing Science, www.liacs.nl/home/ tmoerl/privtech.pdf

[6] [6] Silman, J., "Steganography and Steganalysis: An Overview", SANS Institute, 2001

[7] [7] Jamil, T., "Steganography: The art of hiding information is plain sight", IEEE Potentials, 18:01, 1999

[8] [8] Wang, H & Wang, S, "Cyber warfare: Steganography vs. Steganalysis", Communications of the ACM, 47:10, October 2004

[9] [9] Anderson, R.J. & Petitcolas, F.A.P., "On the limits of steganography", IEEE Journal of selected Areas in Communications, May 1998