Concealing Intelligence in images using LSB

Authors: Mohammad Hashir Khan, Hanshu Agrahari, Ayush Sharma , Ankit Tiwari

Guided By: Ms. Swati Sheoran

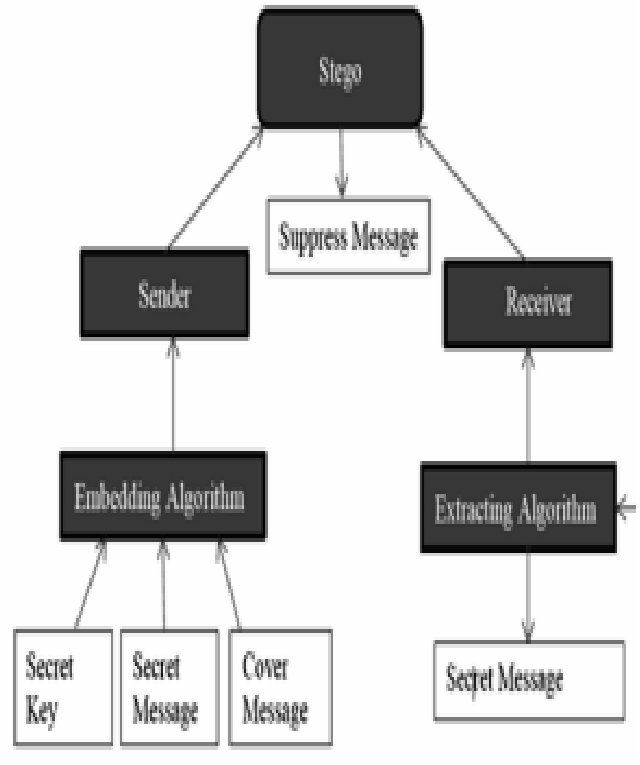
Affiliation: Department of Computer Science and Engineering,

SRM Institute of Science and Technology, Delhi NCR, India

Abstract

Steganography is the art of concealing Intelligence in the cover information in such a way that their existence is unknown. Digital imaging steganography realizes the potential of protecting communication, which is important in most applications today. Steganography has many useful applications. The remarkable development in computing power and security knowledge has made it a leader in security today. The main difficulty in proposing steganography is to balance the possibility of more placement, imperceptibility and security, so it is different from other systems such as cryptography and watermarking. This paper provides a comprehensive state-of-the-art review and analysis of some new steganography techniques. Steganography, by contrast, conceals the existence of communication itself by embedding data within media such as images, audio, or video.

INDEX TERMS: Image Steganography, Information hiding, Image data hiding



Introduction

In today's world, the communication is the basic necessity of every growing area. Everyone wants the secrecy and safety of their communicating data. In our daily life, we use many secure pathways like internet or telephone for transferring and sharing information, but it's not safe at a certain level. In order to share the information in

a concealed manner two techniques could be used. These mechanisms are cryptography and steganography. In cryptography, the message is modified in an encrypted form with the help of encryption key which is known to sender and receiver only. The message cannot be accessed by anyone without using the encryption key. However, the transmission of encrypted message may easily arouse attacker's suspicion, and the encrypted message may thus be intercepted, attacked or decrypted violently. In order to overcome the shortcomings of cryptographic techniques, steganography techniques have been developed. Steganography is the art and science of communicating in such a way that it hides the existence of the communication. Thus, steganography hides the existence of data so that no one can detect its presence. In steganography the process of hiding information content inside any multimedia content like image, audio, video is referred as a —Embedding‖. For increasing the confidentiality of communicating data both the techniques may be combined.

Literature Survey

Steganography has evolved significantly over the years, finding applications across various domains due to its ability to conceal information within multimedia files. Traditional methods such as the LSB technique, which involves modifying the least significant bits in image pixels, are widely employed for their simplicity and effectiveness [2]. This technique is particularly effective in spatial domains, where hidden data remains relatively undetectable to the human eye and minimal file size changes ensure efficient data concealment. However, LSB's limitations include susceptibility to common image manipulations like compression or filtering, which may expose or alter the embedded data [2].

More advanced methods, including DWT and DCT-based steganography, aim to address these limitations by working in the frequency domain, which offers greater robustness against image modifications [1]. DWT-based steganography leverages frequency transformations to conceal data in specific frequency bands, resulting in higher resistance to data loss and distortion during compression and scaling processes [1]. DCT, another frequency-domain technique, is commonly used in JPEG image compression and enhances steganographic resistance to tampering by embedding data in non-perceptible image components [3]. Studies show that frequency-domain techniques generally provide stronger data security, though they may involve higher computational costs compared to spatial domain techniques [3].

Recent research also explores the application of steganography in sectors that require stringent security protocols. For instance, in the medical field, patient data can be embedded within diagnostic images, like X-rays or MRIs, ensuring data confidentiality while adhering to healthcare privacy regulations [1]. Furthermore, steganography has gained attention in military and intelligence applications, where it provides covert channels for communication in surveillance-heavy environments [4]. This includes embedding sensitive data within everyday multimedia files, enhancing security by reducing detectability [4].

In addition to these techniques, current trends in steganography research explore the integration of artificial intelligence and machine learning to improve the detection of steganographic content in digital media. These advancements aim to strengthen security by detecting hidden information in image files, enhancing steganalysis capabilities, and identifying potential vulnerabilities in steganographic algorithms.

Overall, steganography continues to evolve as a critical tool for secure

communication across various fields, offering innovative solutions for covert data embedding and retrieval while addressing the challenges associated with data integrity, efficiency, and detectability.Applications

- Covert Communication: Using steganography does not support covert communication and thus prevents analysis of the sender, message and receiver. Secret files, plans or other sensitive information can be leaked without any warning to the attackers.
- Certain content can be added to the image, such as the name of the person in the image or the location on the map. Copy the steganographic image, also print out all embedded features and display only those features that have the key to determine the steganography can be removed.
- Copyright Protection: A copy protection system that prevents data (usually digital data) from being copied.

Related Work

Steganography, the science of hiding information within seemingly innocuous media, has been a crucial topic of research for secure communication, data protection, and digital rights management. Over the years, various methods have been developed to enhance the robustness, security, and imperceptibility of hidden data. This section explores key contributions to the field, focusing on traditional and contemporary techniques.

1. Classical Steganography Techniques

One of the foundational methods in digital steganography is the Least Significant Bit (LSB) embedding technique. It involves replacing the least significant bits of the carrier media (often images) with the secret data. LSB embedding is simple and computationally efficient, but it is highly susceptible to steganalysis due to detectable changes in the media. Yang et al. (2008) proposed an improvement by adapting data hiding to edge areas in images, enhancing security by minimizing visible distortions. While LSB is effective in low-risk environments, modern requirements for security have pushed the development of more complex techniques.

2. Transform Domain Techniques

To overcome the limitations of spatial domain methods like LSB, transform domain techniques such as the Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT) are commonly used. These methods embed data in frequency components of the carrier media, making it more resistant to common image processing operations like compression and noise addition. Research by Saidi et al. (2017) showed that DCT combined with chaotic maps offers improved robustness and security. Similarly, Kumar and Kumar (2018) proposed a modified DWT technique, which further enhances robustness against image manipulations.

3. Audio and Video Steganography

Beyond images, steganography in audio and video files has also seen significant progress. Techniques like spread spectrum and phase coding are used for embedding data in audio files. These methods are designed to resist steganalysis by distributing the hidden information across multiple frequencies or altering phase information, making detection more difficult. Video steganography typically utilizes motion vectors or redundant frames to hide data, taking advantage of the large amount of data present in video streams. Wang et al. (2021) explored payload location optimization in JPEG images, which has direct applications in video steganography.

4. Recent Advances

Recent work in steganography includes the use of deep learning and neural networks to automate the embedding and extraction of hidden data. This approach allows for more sophisticated and flexible steganography systems. Neural networks can be trained to find optimal regions in a carrier file where data can be embedded with minimal detection risk. Islam et al. (2018) demonstrated how neural networks could enhance the robustness of image watermarking techniques, which are closely related to steganography.

Another recent trend is the integration of blockchain technology with steganography. As noted by contemporary researchers, blockchain's decentralized and secure nature makes it a promising platform for steganography, particularly in applications requiring immutable and secure data transfer. For instance, businesses are investigating blockchain for secure data exchange and supply chain traceability.

5. Steganalysis and Countermeasures

With the growing sophistication of steganographic techniques, steganalysis—the process of detecting hidden data—has also advanced. Techniques like statistical analysis and machine learning have become popular in detecting inconsistencies in carrier files that suggest the presence of hidden data. Researchers have developed robust steganalysis tools to counteract the increasing complexity of steganographic methods. These tools often analyse anomalies in the statistical distribution of pixel values or audio frequencies, making it easier to detect hidden information.

Steganography's effectiveness depends on striking a balance between security and detectability. Recent research by Luo et al. (2017) focused on reversible data hiding methods that allow the original carrier to be restored after the embedded data is extracted, minimizing alterations to the carrier file

Furthermore, Shafi et al. (2018) proposed an adaptive hybrid fuzzy-wavelet approach that optimizes the bit reduction and pixel adjustment to enhance the quality of steganography in various media formats.

6. Challenges and Future Directions

Despite the significant progress in steganography, challenges remain. Maintaining a high data capacity without compromising the imperceptibility of the hidden data is a primary concern. Techniques that alter fewer bits in the carrier media are less detectable but often limit the amount of data that can be embedded. Furthermore, standardization and regulation of steganographic methods are necessary to ensure their responsible use, particularly in preventing misuse in cyberattacks.

The future of steganography may involve greater integration with emerging technologies like quantum computing, which has the potential to either break existing steganographic methods or offer new ways to hide data. Additionally, the development of steganography in cloud computing and IoT environments is expected to address the growing need for secure communication in distributed systems.

STEGANOGRAPHY TECHNIQUES

Below are the categories of steganography technologies:

Frequency Domain Technology

In this technology, various algorithms and modifications are used to hide the messages, where the embedding method is suggested

According to many Algorithms, this technique is a bit tedious and is classified as follows: -

Discrete Cosine Transform Technology

It is used to convert the signal to the fundamental frequency using the Discrete Cosine Transform (DCT) properties.

Discrete Wavelet Transform Technology

When the wavelet is detected discretely, it is a discrete wavelet transform (DWT). frequency component of the pixel value.

Spatial domain method

In this method, a few bits of the image pixels are directly changed to hide the data. This process is classified as:

Pixel Value Difference

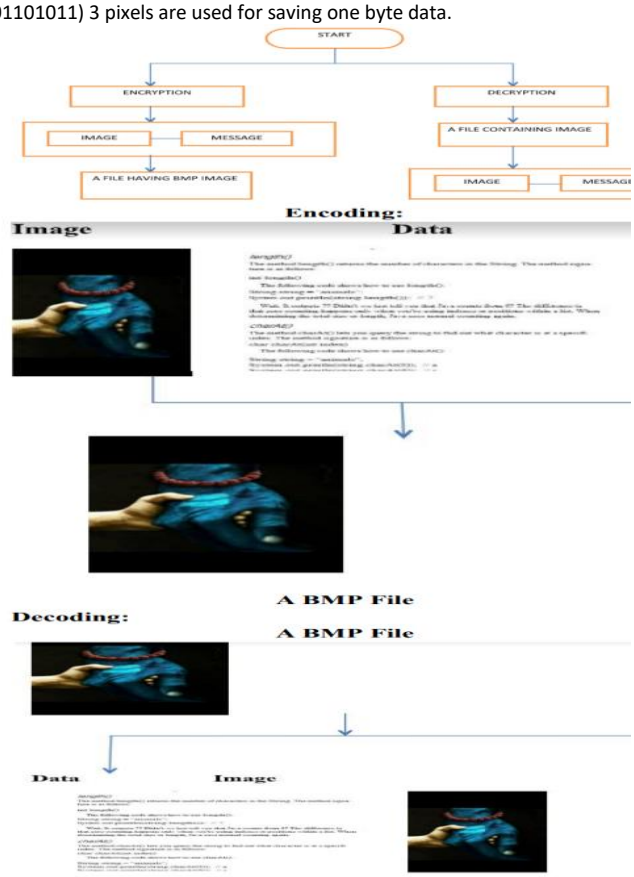
In this process, quantization multiple tables are created, the payload is determined and the countability of the steganography is maintained.

Edge-based data embedding method

In this method, every pixel edge in the image is used. First, we calculate the mask image and determine the edge pixels with the edge detection method. Information is hidden in the LSB bits of the edge pixels and the receiver receives the steganographic bits. For example, the key in binary number: 110100101001, is the LSB of 1 on the right. The secret message is in the LSB of the image.

Bitmap Steganography Technique

One of the simplest type of picture is the Bitmap type as for decreasing file size it have no technologies. A bitmap image created from pixels is the structure of these files, three colours (green, red and blue say GRB) are used for pixel creation, one byte information is contained in every colour of pixel and it shows the colour's destiny. The colours which we see in these pictures are made by merging these three colours. 1 Byte is equivalent to 8 bit, and the first bit is called as Most-Significant-Bit (MSB) and last bit is called LeastSignificant-Bit (LSB), now here for writing security information in BMP picture we use LSB bit. Now if the (8st layer) that is the last layer of information is need to be changed then we only need to change last bits of pixel,3 bits are there in each pixel so the bits memory for writing our data is equivalent to 3*height*width. Data name and data file name should be written properly and it can be done by assigning first bit of memory. (01110101 01010101 11101100) (11010010 10010101 00010100) (10110010 10011100 01101011) 3 pixels are used for saving one byte data.



Methodology

This research proposes a simpler adaptive method that effectively determines the optimal pattern for inverted Least Significant Bit (LSB) substitution in steganography. The primary objective is to enhance the embedding process by evaluating error ratios associated with various patterns before embedding messages into a container image. Steganography, the practice of concealing messages within other non-secret data, primarily involves two critical processes: embedding and extracting messages.

Embedding Process

The embedding process, as illustrated in the accompanying diagrams, follows a systematic approach:

Reading Pixel Values: The initial step involves reading the pixel values of a container image, along with determining its dimensions, denoted as m*m* for width and n*n* for height. For example, a sample image matrix might consist of pixel values: [228 233 162 140 231 33 25 45]

Pixel Count and Array Conversion: The total number of pixels is calculated by multiplying m*m* and n*n*. The image is then converted into a one-dimensional array, saved in a variable C*C*. For instance, with m=4*m*=4 and n=2*n*=2, the array C*C* becomes: C=[228,231,233,33,162,25,140,45]*C*=[228,231,233,33,162,25,140,45]

Message Preparation: The digital message intended for embedding is read, ensuring its size fits within the container image. The message is converted into its ASCII representation and subsequently into binary form. For example, the character 'A' translates to the ASCII value 65, which in binary is represented as M=11001010*M*=11001010.

Message Encryption: A key for message encryption is entered, and the message is encrypted using the RC4 algorithm, producing ciphertext. For example, if the key is 'password', the encrypted message for M=65*M*=65 may yield a binary equivalent of 1100101011001010.

Pattern Selection: The next step involves selecting a three-bit combination from the pixel values, such as the 6th, 7th, and 8th bits.

Error Counting: Sixteen variables are established to count eight patterns, each representing the number of pixels that change their values during the embedding process. The remaining eight variables count those that remain unchanged. The number of bits that change (denoted as p*p*) and those that do not change (denoted as p'*p*') are calculated using specific equations.

Error Calculation: The total error for the eight patterns is summed, stored in an error variable e*e*. The smallest total error is computed, guiding the selection of the optimal embedding pattern.

Pattern Embedding: Steps 3 to 8 are repeated for other three-bit combinations, resulting in a total of 21 combinations. The combination yielding the smallest error is selected for embedding, and the corresponding pattern data is saved as an extraction key k*k*.

Finalizing the Stego Image: The binary values of the modified pixels are converted back to decimal form, reshaping the stego array to match the original image dimensions m×n*m*×*n*. The resulting stego image preserves the original pixel values while embedding the secret message.[3]

Extraction Process

Once the embedding process is completed, the receiver can extract the hidden message using the following steps:

Reading the Stego Image: The receiver reads the stego image and retrieves its dimensions.

Pixel Reshaping: The stego image is reshaped into an array, stored in variable S*S*.

Extraction Key Utilization: The extraction key k*k* is read to identify the bit combination and inverted bit pattern used during embedding.

Message Extraction: The LSBs of the stego image are read based on the extraction key, allowing the recovery of the embedded message bits.

Binary to ASCII Conversion: The extracted bits are grouped and converted back into an ASCII number, revealing the encrypted message.

Message Decryption: Finally, the message is decrypted using the RC4 algorithm with the same key, resulting in the original ASCII value, which is then converted back to its character form, completing the extraction process.

This method demonstrates a robust and efficient approach to

steganography, enhancing both the security and reliability of message embedding and extraction while minimizing detectable alterations to the carrier image.[3]

Discussion

The adaptive LSB approach presented in this study enhances both security and detectability reduction. Traditional LSB methods faced challenges in high-security contexts due to detectable pattern changes; however, the adaptive pattern selection approach here addresses this by dynamically choosing the least disruptive bit combinations. By optimizing based on error ratios, the method reduces the visibility of alterations, making detection by steganalysis tools significantly more challenging.

The use of RC4 encryption adds an additional layer of security to the embedded message, mitigating risks even if data extraction is partially successful. This dual approach, combining adaptive LSB embedding with encryption, balances the need for high data capacity with minimal detectability. Despite its robustness, future iterations could benefit from integrating artificial intelligence to dynamically adapt patterns based on media type and complexity.

Furthermore, incorporating blockchain or cloud environments may improve traceability and secure distribution in networked applications. While the current approach is highly effective, addressing its dependency on image resolution and size could broaden applicability across diverse media.

Conclusion

The novelty of this research is in the use of the adaptive pattern to perform the inverted LSB. Before the message is embedded, the message and container image bits are measured and the error ratio for each bit combination is calculated. There are eight patterns (000 to 111) that are used to invert the LSB for each combination, which are 2-bits + the LSB of the container image pixels. Inverted LSB is performed To get a smaller error ratio for each pattern. All smaller error ratios of each pattern are summed for each bit combination. The bit combination that produces the smallest error ratio is chosen to embed the message. Since it is based on the measurement of the error ratio, this best bit combination may differ for a different couple of container image and message size. Testing to standardized and medical less heterogeneous images shows that the proposed method is successful in increasing the imperceptibility based on PSNR and SSIM. In subsequent research, this method can be developed with the addition of parameters to determine patterns and to do optimization using artificial intelligence methods.

Datasets

- https://images.app.goo.gl/9duZhuyFEjiQd4es7
- https://images.app.goo.gl/EQC9f1E8WJ6ktsx29
- https://images.app.goo.gl/xFH6RmfYXnHCEucp6

References

[1] Ansari, A.Q.; Khan, M.E.; Pant, M., "Data Security by Steganography: A Review," ResearchGate, vol., no., pp., 2019.https://www.researchgate.net/publication/379806300_Data_security_by_steganography_A_review

[2] Sharma, S.; Parashar, P., "Steganography in Images Using LSB Technique," ResearchGate, vol., no., pp., 2020. https://www.researchgate.net/publication/371671984_Steganography_in_Images_Using_LSB_Technique#:~:text=DWT)%20based%20steganography.-,The%20LSB%20algorithm%20is%20implemented%20in%20spatial%20domain%20in%20which,the%20frequency%20domain%20and%20the

[3] Thampi, S.M.; Mukhopadhyay, S.; Moschoyiannis, S., "A Survey on Image Steganography," IEEE Explore, vol., no., pp., 2015. https://ieeexplore.ieee.org/document/7226122

[4] Changder, S.; Chakraborty, M.; Sarkar, R., "Steganography Techniques and Their Applications in Security," IEEE Explore, vol., no., pp., 2001. https://ieeexplore.ieee.org/document/959097

[5] Fridrich, J., Goljan, M., Du, R., 2001. Detecting LSB steganography in color and gray scale images. IEEE Multimed. 8, 22–28. https://doi.org/10.1109/93.959097.

[6] Supriadi Rustadǂ, De Rosal Ignatius Moses Setiadi, Abdul Syukur, Pulung Nurtantio Andono, Inverted LSB image steganography using adaptive pattern to improve imperceptibility. https://doi.org/10.1016/j.jksuci.2020.12.017.