International Journal of Advanced Multidisciplinary Research and Educational Development
Volume 2, Issue 2 | March – April 2026 | www.ijamred.com
ISSN: 3107-6513

Hidden Data Communication Using Image Steganography

Samvrutha P S,Dr.M.Usha Devi
[1]B.Sc Digital & Cyber Forensics Science, Rathinam College of Arts and Science, Coimbatore - 641021, India.

Abstract

The increasing use of digital communication has made secure data transmission a critical requirement. Traditional encryption techniques protect message content but fail to conceal the existence of communication itself, making them susceptible to detection and interception. This paper presents a system for Hidden Data Communication using Image Steganography, which embeds secret messages within digital images using the Least Significant Bit (LSB) technique. The system is designed with a modular architecture incorporating encoding, decoding, input validation, and password-based security within a single graphical user interface developed using Python with Pillow and Tkinter libraries. Experimental results demonstrate that the system effectively hides and retrieves data without visually altering the cover image, providing an additional layer of security over conventional encryption. The proposed approach offers a simple, cost-effective, and user-friendly solution suitable for secure covert communication in academic and practical applications.

Key Words: *Image Steganography, LSB Technique, Data Hiding, Secure Communication, Python, Pillow, Tkinter, Covert Communication, Digital Image Processing*

1. INTRODUCTION

The rapid growth of digital communication systems has significantly increased the volume of sensitive data transmitted over public networks. This has resulted in an escalating demand for robust security mechanisms that not only protect the content of communications but also conceal their very existence. Steganography, derived from the Greek words meaning 'covered writing', addresses this need by embedding secret information within ordinary-looking digital media such as images, audio, or video files without arousing suspicion.

Image steganography is among the most widely studied forms of steganography due to the high redundancy present in digital images. The human visual system is unable to perceive minute changes in pixel values, making images an ideal carrier medium for hidden data. The Least Significant Bit (LSB) technique is one of the most commonly employed methods, as it modifies only the least significant bits of pixel values, ensuring that visual quality is preserved while data is securely embedded.

Unlike encryption, which transforms data into an unreadable format that still signals the presence of secret information, steganography conceals the fact that any communication is taking place at all. This dual-layer approach, when both techniques are combined, significantly strengthens the security posture of a communication system.

Existing steganography tools, however, suffer from notable limitations including poor usability, lack of integrated security features such as password protection, and limited support for graphical user interfaces. These deficiencies reduce their suitability for practical demonstration and real-world deployment. To address these gaps, this paper proposes a comprehensive Hidden Data Communication system using Image Steganography that integrates encoding, decoding, and password-based access control within a single, user-friendly Python application.

2. LIMITATIONS OF EXISTING SYSTEMS

Existing data communication security systems rely predominantly on cryptographic techniques to protect information. While these methods are effective in securing message content, they present several limitations that reduce their effectiveness in practical environments.

The foremost limitation is the lack of data concealment. Encrypted messages, though unreadable, remain visible to network observers and may attract targeted attacks. The presence of ciphertext signals the existence of sensitive communication, making the channel a potential target for interception or brute-force attempts.

Additionally, encryption systems require robust key management infrastructure. Users must securely generate, distribute, and store cryptographic keys, introducing operational complexity and increasing the risk of key compromise. If a key is exposed, the entire security of the communication is undermined.

Existing steganography tools also have significant usability drawbacks. Most tools operate through command-line

ISSN: 3107-6513

interfaces and require technical expertise to use, limiting their accessibility to non-technical users. Furthermore, these tools typically lack integrated security mechanisms such as password protection, and do not provide end-to-end functionality within a single platform. This fragmentation increases operational complexity and reduces their effectiveness for demonstration and deployment purposes.

3. PROPOSED SYSTEM

To overcome the limitations of existing systems, this paper proposes a Hidden Data Communication system using Image Steganography. The proposed system provides a secure, integrated, and user-friendly approach for embedding and retrieving secret messages within digital images. It combines data hiding with password-based access control to ensure both concealment and security.

The system employs the Least Significant Bit (LSB) technique to embed the binary representation of a secret message into the pixel values of a cover image. For each pixel, the least significant bit of each colour channel (Red, Green, Blue) is replaced with one bit of the message. This process introduces minimal changes to the pixel values, ensuring that the visual quality of the image is maintained and the modification remains imperceptible to the human eye.

A unique end marker ('11111110') is appended to the binary message to signal the end of the hidden data during extraction. This allows the decoding module to accurately identify and retrieve only the embedded content without processing the entire image.The system is implemented using Python 3 with the Pillow (PIL) library for image processing and Tkinter for graphical user interface development. Both libraries are open-source and freely available, making the solution cost-effective and accessible. The application runs on standard Windows hardware without requiring specialised resources.

Password protection is integrated directly into the decoding process. A fixed password is verified before the hidden message is revealed, ensuring that only authorised users can access the embedded data. This adds a critical security layer that compensates for the inherent detectability risk of steganographic images.

4. SYSTEM ARCHITECTURE

The system architecture of the proposed solution follows a modular design, where each component is responsible for a specific function. The overall workflow proceeds through four principal stages: input handling, encoding, decoding, and output generation.

In the input stage, the user provides a secret message and a password through the graphical user interface. The input module validates these entries to ensure that neither field is empty before further processing. Once validated, the message is passed to the encoding module.

The encoding module converts the text message into its binary representation using ASCII encoding. Each character is transformed into an 8-bit binary string, and an end marker is appended. The binary data is then sequentially embedded into the least significant bits of the RGB channels of each pixel in the cover image, proceeding row by row until all data has been stored. The resulting encoded image is saved as an output PNG file.

During decoding, the system reads the least significant bits from each colour channel of the encoded image's pixels and reconstructs the binary string. The binary data is segmented into 8-bit groups and converted back to characters until the end marker is detected, yielding the original message. Before displaying the retrieved message, the security module verifies the entered password. If authentication fails, access is denied and an error notification is displayed.

The output module presents results through message boxes in the graphical interface and stores the encoded image in the project directory for future use. This modular design ensures separation of concerns, simplifying maintenance and enabling future enhancements.

5. METHODOLOGY

The methodology of the proposed system follows a structured pipeline designed for reliability, accuracy, and usability. The key phases are described below.

Phase 1 – Environment Setup: The development environment is configured on a Windows operating system with Python 3.x installed. The required libraries, Pillow for image processing and Tkinter for GUI development, are installed using the Python package manager. No specialised hardware or licensed software is required.

Phase 2 – Message Encoding: The user inputs a secret message and selects a cover image through the GUI. The message is converted to a binary string, and an end marker is appended. The binary data is embedded into the LSBs of the image pixels sequentially, and the modified image is saved to disk as output.png.

Phase 3 – Message Decoding: The user selects the encoded image and provides the password. The system verifies the password and, upon successful authentication, extracts the binary data from the image's LSBs. The binary string is decoded to recover the original message, which is then displayed to the user.

Phase 4 – Validation and Testing: The system is tested using various input messages and image types. Functional testing validates correct encoding and decoding behaviour. Error handling tests ensure that invalid inputs, incorrect passwords, and empty fields are managed gracefully. Visual inspection confirms that encoded images are indistinguishable from the originals.

6. RESULTS AND DISCUSSION

The proposed system was implemented and tested under different operational scenarios. The results confirm that the system performs encoding and decoding operations accurately while maintaining the visual integrity of the cover image.

During the encoding process, the system successfully embedded text messages of varying lengths into PNG images. Visual comparison between original and encoded images revealed no perceptible differences, validating the effectiveness of the LSB technique in preserving image quality. The encoded images were stored as output.png files with the hidden data intact.

The decoding process demonstrated accurate retrieval of the original message when the correct password was provided. When an incorrect password was entered, the system correctly denied access and displayed an appropriate error message, confirming the reliability of the password protection mechanism.

Table 1: System Test Results

Test Scenario	Description	Observation
Encoding – Short Message	Message embedded into PNG cover image	Encoded successfully; no visible distortion
Decoding – Correct Password	Decoding attempted with valid password	Original message retrieved accurately
Decoding – Wrong Password	Decoding attempted with invalid password	Access denied; error message displayed
Empty Input Validation	Message or password field left empty	System prompts user to complete the required field

The system maintained stable performance across all test cases with no significant resource overhead. The lightweight design ensures that encoding and decoding operations are completed efficiently on standard computing hardware, confirming the operational feasibility of the proposed approach.

7. CONCLUSIONS

This paper presented a Hidden Data Communication system using Image Steganography, designed to provide secure, covert transmission of information within digital images. The system employs the Least Significant Bit technique to embed and extract secret messages without perceptible modification to the cover image, addressing the primary weakness of traditional encryption-only approaches, which expose the existence of sensitive communication.

The proposed system demonstrates that practical steganography tools can be developed using freely available, open-source technologies at minimal cost. The integration of encoding, decoding, and password-based security within a single graphical interface significantly improves usability compared to existing command-line-based tools. Experimental evaluation confirms accurate message retrieval, effective password enforcement, and robust input validation.

Future enhancements may include support for multiple image formats such as JPEG and BMP, integration of advanced cryptographic techniques alongside steganography for layered security, increased data capacity through adaptive embedding algorithms, and extension of the system to support audio and video steganography. Real-time processing capabilities and network transmission support could further extend the system's applicability to enterprise-level secure communication environments.

ACKNOWLEDGEMENT

The author expresses sincere gratitude to Dr. M. Usha Devi, Assistant Professor, Department of Computer Science, Rathinam College of Arts and Science, Coimbatore, for her valuable guidance, supervision, and encouragement throughout the development of this project. The author also acknowledges the support of Dr. T. Velumani, Head of the Department, and the entire faculty of the Department of Computer Science.

REFERENCES

[1] N. F. Johnson and S. Jajodia, "Exploring Steganography: Seeing the Unseen," IEEE Computer, vol. 31, no. 2, pp. 26–34, 1998.

[2] R. J. Anderson and F. A. P. Petitcolas, "On the Limits of Steganography," IEEE Journal on Selected Areas in Communications, vol. 16, no. 4, pp. 474–481, 1998.

[3] A. Cheddad, J. Condell, K. Curran, and P. McKevitt, "Digital Image Steganography: Survey and Analysis of Current Methods," Signal Processing, vol. 90, no. 3, pp. 727–752, 2010.

[4] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for Data Hiding," IBM Systems Journal, vol. 35, nos. 3–4, pp. 313–336, 1996.

[5] Python Software Foundation, Python 3 Documentation, 2024. Available: https://docs.python.org/3/

[6] Pillow (PIL Fork) Documentation, 2024. Available: https://pillow.readthedocs.io/

[7] NIST, "Framework for Improving Critical Infrastructure Cybersecurity," National Institute of Standards and Technology, 2018.

BIOGRAPHY

Samvrutha P S is currently a B.Sc. Digital and Cyber Forensics Science student at Rathinam College of Arts and Science, Coimbatore, India. Her areas of interest include image steganography, data security, digital forensics, and Python-based application development.