SIDDAGANGA INSTITUTE OF TECHNOLOGY, TUMAKURU- 3

(An Autonomous institution affiliated to Visvesvaraya Technological University- Belagavi, Approved by AICTE,
Accredited by NAAC with ‘A++’ Grade, Awarded Diamond College Rating by QS I-GAUGE & ISO 9001:2015 certified)

MINI PROJECT REPORT

ON

“DIGITAL STEGANOGRAPHY”

submitted in the partial fulfilment of the requirements for VI semester,
Bachelor of Engineering in Computer Science and Engineering

By

Dhanaraj Chandrashekhar Nandikoppa	**1SI20CS032**
Mahesh G	**1SI20CS056**

Under the guidance of

Mr. Gururaj S P **M.Tech**
Assistant Professor

Department of Computer Science and Engineering

(Program Accredited by NBA)

Academic Year: 2022-23

Siddaganga Institute of Technology, Tumakuru-3

(An Autonomous institution affiliated to Visvesvaraya Technological University- Belagavi, Approved by AICTE, Accredited by NAAC with 'A++' Grade, Awarded Diamond College Rating by QS I-GAUGE & ISO 9001:2015 certified)

Department of Computer Science and Engineering

(Program Accredited by NBA)

CERTIFICATE

This is to certify that the mini project entitled "Digital Steganography" is a bonafide work carried out by **Dhanaraj Chandrashekhar Nandikoppa (1SI20CS032), Mahesh G (1SI20CS056)** of VI semester **Computer Science and Engineering**, **SIDDAGANGA INSTITUTE OF TECHNOLOGY** during the academic year 2022-2023.

Signature of the Guide
Mr. Gururaj S P M.Tech
Assistant Professor

Signature of the Convener
Thejaswini S M.Tech
AssistantProfessor

Signature of the HOD
Dr. A S Poornima
Prof. and Head, Dept. of CSE

Name of the Examiners: **Signature with Date**

1. **Prof.**

2. **Prof.**

ACKNOWLEDGEMENT

With great reverence, we express our sincere gratitude and salutations to his holiness **Dr. Sree Sree Shivakumara Swamigalu,** Founder of Sree Siddaganga Education society and **Sree Sree Siddalinga Swamigalu,** President of Sree Siddaganga Education society for their blessings.

First and foremost, we wish to express our deep sincere feelings of gratitude to our institution, **Siddaganga Institute of Technology,** for providing us an opportunity for completing our project successfully.

We are grateful to **Dr. M. N. Channabasappa,** Director and **Dr. Shivakumaraiah,** CEO, Siddaganga Institute of Technology-Tumakuru, for their kind cooperation and encouragement.

We express our kind thanks to **Dr. S V Dinesh,** Principal, Siddaganga Institute of Technology, Tumakuru for his encouragement towards students attitude.

We express our heartfelt thanks to **Dr. A S Poornima,** Professor and Head, Department of Computer Science and Engineering, Siddaganga Institute of Technology, Tumakuru for her suggestions and advice.

We express our gratitude and humble thanks to our project guide, **Mr. Gururaj S P,** Assistant Professor, Department of Computer Science and Engineering, Siddaganga Institute of Technology, Tumakuru for guiding us to complete our Mini-project successfully.

We thank our mini project convener, **Mrs. Thejaswini S**, Assistant Professor, Department of Computer Science and Engineering, for allowing us to take up this idea as our Mini Project.

We are conscious of the fact that we have received cooperation in many ways from the **Teaching, Technical and supporting staff** of the Department of Computer Science and Engineering and we are grateful to all for their cooperation.

We express heartfelt gratitude to our **Parents and Friends** for their constant moral support and encouragement throughout this work.

ABSTRACT

Digital steganography is a fascinating field that explores the art of hiding sensitive information within multimedia files, enabling covert communication and secure data transmission. This project aims to develop a robust and efficient digital steganography system capable of concealing confidential data within various types of multimedia content, including images, audio files, and videos.

The project focuses on the implementation and evaluation of advanced steganographic techniques to ensure high concealment capacity and minimal perceptibility. The chosen techniques involve modifying the least significant bits of pixel values in images exploiting the redundancies and imperceptible changes in the media.

Overall, this project aims to contribute to the field of digital steganography by developing an efficient, secure, and user-friendly system that enables the covert communication and secure transmission of sensitive data within multimedia files. The results and findings of this project can have significant implications in areas such as secure communication, copyright protection, and data

Table of Contents

CHAPTER 1

INTRODUCTION

Digital steganography is the practice of concealing a message or information within a digital file without changing its external appearance. It involves embedding hidden data within the bits of a carrier file, such as an image, audio, or video file, in a way that is undetectable to the human eye or ear. The goal of steganography is to hide the existence of the message, rather than the content of the message itself.

Steganography has been used for centuries to hide secret messages within written or printed documents, but in the digital age, it has become a popular tool for both legitimate and nefarious purposes. Digital steganography is used in a variety of applications, such as watermarking images, hiding metadata in files, and concealing sensitive information in communications.

Steganography can be used for both benign and malicious purposes. For example, it can be used to protect confidential information, to embed copyright or ownership information in media files, or to hide messages in plain sight. On the other hand, it can also be used by cybercriminals and terrorists to communicate covertly, to hide malware or other malicious payloads, or to steal sensitive information.

The methods used in digital steganography are constantly evolving, and new techniques are being developed to hide information more effectively. As a result, steganography is an area of active research in the fields of computer science and information security.

CHAPTER 2

LITERATURE SURVEY

- Alvin, A. Wicaksana and M. I. Prasetiyowati, **"Digital Watermarking for Color Image Using DHWT and LSB,"**

  The paper proposes using digital watermarking to protect against copyright infringement of digital content. The method uses DHWT and LSB steganography to embed a watermark into digital images. Testing showed high PSNR results for images of different resolutions.

- D. Bhattacharyya and A. Haveliya, **"A Robust Method for Data Hiding via Combination of Colour Images and PDF Files,"**

  The paper proposes a methodology for embedding color image data into a PDF file and vice versa. The signature image is encrypted and embedded into the host image's red components without requiring knowledge of the original image for recovery. The decryption algorithm reconstructs the original signature file by decrypting the password and secret message from the stego-file. MATLAB is used to simulate the data embedding process, which shows no visible distortions in the host image.

- M. Ye, F. Liu, C. Yang and X. He**, "Steganalysis Based on Weighted Stego-Image for 2LSB Replacement Steganography,"**

  This paper proposes a new weighted stego-image model for 2LSB replacement steganography and a new steganalytic method based on this model. The method estimates the embedding ratio of hyper-LSB by introducing a proposed LSB detection method and calculates the embedding ratio of LSB based on the weighted stego-image model. Experimental results demonstrate that the new method is effective.

- Premalatha P and Amsaveni A, **"Data hiding in a digital image with FPGA implementation,"**

  This paper discusses a low-complexity steganography system with digital images as the host signal, using an LSB-based approach to hide embedded information. The paper proposes a Simulink block for the embedding and extraction processes, which can be converted into VHDL code and implemented in an FPGA. The aim is to hide the existence of the embedded information to prevent attacks.

- O. Evsutin, A. Melman and R. Meshcheryakov, **"Digital Steganography and Watermarking for Digital Images: A Review of Current Research Directions,"**

  This paper provides an overview of steganography and watermarking methods for solving information security tasks related to multimedia data, such as protecting against leakage of confidential information and copyright protection. The paper discusses the main applications of these methods and reviews current trends in the development of algorithms for data hiding in digital images. The review is focused on contemporary works illustrating current research directions and identifies problems in the field of digital steganography and digital watermarking.

CHAPTER 3

3.1 PROBLEM STATEMENT

The problem addressed by digital steganography is the need to protect sensitive or confidential information from being intercepted or monitored by third parties, whether intentionally or inadvertently. With the increasing reliance on digital communication and information sharing, it has become more important than ever to develop effective methods of securing and sharing information. Digital steganography provides a powerful solution by allowing users to hide messages within digital files such as images or audio files in a way that is not detectable by unauthorized parties. However, the effectiveness and security of digital steganography systems can vary depending on the specific techniques and algorithms used. Therefore, there is a need to develop reliable and robust steganography systems that can be used in a variety of different contexts and applications, from personal messaging to intelligence operations. The goal of this project is to develop a digital steganography system using Java that can provide high levels of security, accuracy, and usability.

3.2 OBJECTIVES

The main objective of Digital steganography includes:

- Develop a steganography system using Java programming language
- Implement common steganography techniques such as Least Significant Bit (LSB) and Discrete Cosine Transform (DCT)
- Evaluate the performance and security of the implemented system through testing and analysis
- Explore potential improvements to the steganography system, such as the use of encryption or more sophisticated algorithms
- Create a user-friendly interface for the steganography system to allow for easy use and accessibility by a wide range of users
- Document the development process, including design decisions, implementation details, and testing procedures, to allow for future maintenance and improvement of the system.

CHAPTER 4

SYSTEM DESIGN

An architectural diagram in fig 4.1 for a digital steganography project provides an overview of the system's components and their relationships. It includes the user interface, encoding/decoding module, file management, and encryption/decryption components. The encoding/decoding module performs steganography processes, including embedding and extracting hidden messages. The file management component manages the files being modified by the system, including selecting files and ensuring file integrity. The encryption/decryption component provides additional security. Overall, the architectural diagram illustrates how the componentsof the system work together to achieve steganography functionality.

In our project. The system consists of two main graphical user interface (GUI) screens: "Home" and "ComposePage" (for composing a steganographic message). There is also a "BreakPage" screen for decoding a steganographic message. The UI elements include buttons, text fields, labels, and image display areas.

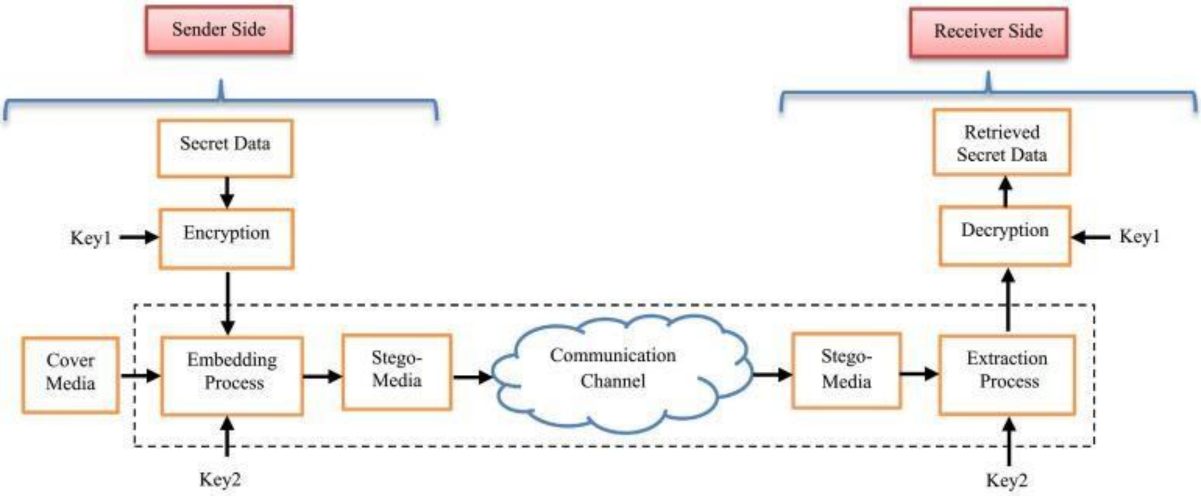


Fig 4.1 : System Architecture

CHAPTER 5

HIGH LEVEL DESIGN

A use case diagram in fig 5.1 for a digital steganography project would depict the different ways in whichusers can interact with the system. It would illustrate various use case scenarios, such as hiding data within an image, retrieving hidden data from an image, configuring steganography settings, authenticating users, generating steganographic keys, and managing steganographic data.

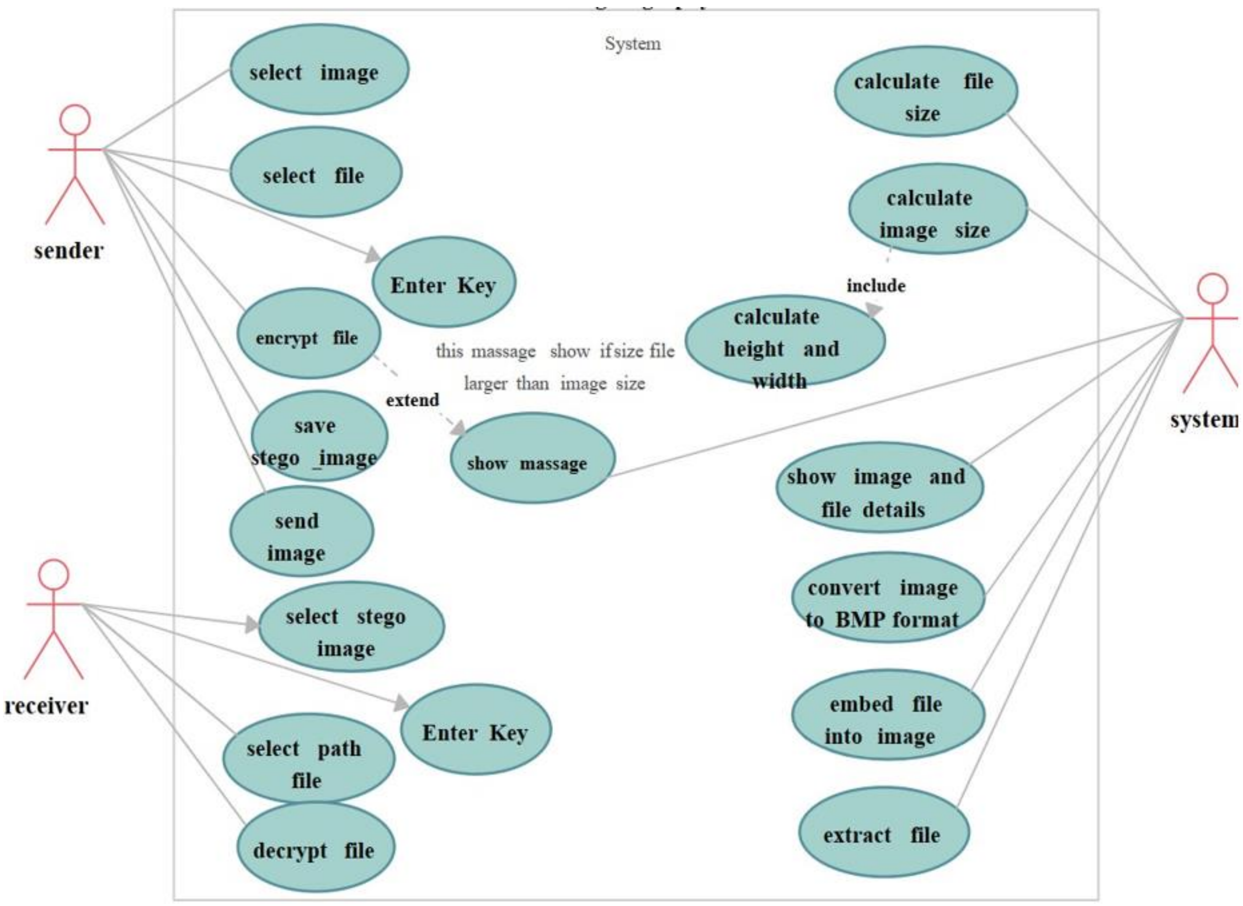


Fig 5.1 : Use case diagram

CHAPTER 6

TOOLS AND TECHNOLOGIES

- **Java Development Kit (JDK):** The JDK is a software development kit that provides tools and libraries necessary for Java development. It includes the Java compiler, runtime environment, and various utilities. To work with Java code, you'll need to install the JDK, which can be downloaded from the official Oracle website. It ensures that you have the necessary components to compile and run Java programs.

- **Integrated Development Environment (IDE):** An IDE is a software application that provides comprehensive tools and features to streamline the development process. Eclipse, IntelliJ IDEA, and NetBeans are popular Java IDEs. IDEs offer features like code editing, debugging, project management, and integrated build tools. They make it easier to write, test, and debug Java code, providing a more productive development environment.

- **Java Swing:** Java Swing is a GUI toolkit for Java applications. It provides a rich set of components and libraries for creating graphical user interfaces. Swing includes various classes for buttons, text fields, labels, menus, and other GUI elements. It also offers layout managers to arrange components within containers. With Swing, you can design and implement the user interface for your steganography system, allowing users to interact with the application.

- **Image Processing Libraries:** To manipulate multimedia files, including images, you may require image processing libraries. Java Advanced Imaging (JAI) and Java Media Framework (JMF) are popular libraries for multimedia processing in Java. JAI provides advanced image processing capabilities like image enhancement, image transformation, and pixel-level manipulation. JMF supports audio and video playback, capture, and editing. These libraries can be utilized to perform image-related operations in your steganography system.

- **Steganography Algorithms:** Steganography algorithms are used to embed and extract data from carrier files (such as images or audio files) while maintaining their visual or auditory integrity. Common steganography techniques include LSB substitution, where data is hidden by modifying the least significant bits of the carrier file's pixels or samples. Other techniques involve frequency domain transformations like Discrete Cosine Transform (DCT) to embed data in the frequency coefficients. You can implement or utilize existing steganography algorithms based on your specific requirements.

- **Encryption Libraries:** If you plan to incorporate encryption for the hidden data, cryptographic libraries like the Java Cryptography Architecture (JCA) or Bouncy Castle can be used. These libraries provide implementations of encryption and decryption algorithms such as AES (Advanced Encryption Standard). You can utilize these libraries to secure the hidden data by encrypting it before embedding and decrypting it during extraction, ensuring the confidentiality and integrity of the information.

- **PixelGrabber:** The PixelGrabber class is part of the java.awt.image package and allows access to individual pixels of an image. It is used in the code to retrieve pixel data from loaded images and manipulate them during the steganographic message hiding and extraction process.

- **FileDialog:** The FileDialog class is part of the java.awt package and provides a dialog box for selecting files from the file system. It is utilized in the code to display a file dialog and enable users to choose image files for loading and saving.

CHPATER 7

IMPLEMENTATION

In image steganography, the least significant bit (LSB) algorithm is a commonly used technique to hide data within the least significant bits of pixel values in an image. The LSB algorithm is a simple and straightforward method that offers a basic level of security and minimal visual distortion. Here's an overview of how the LSB algorithm works:

Image Representation: In digital images, each pixel is typically represented by a combination of red, green, and blue (RGB) color values. Each color value is an 8-bit integer, ranging from 0 to 255, representing different levels of intensity.

Secret Data: The secret data that you want to hide can be in the form of text, binary data, or any other information you wish to conceal within the image.

Pixel Modification: The LSB algorithm involves modifying the least significant bits of the pixel values in the image to encode the secret data. Since the least significant bit has the least impact on the overall color perception, modifying it will produce minimal visual changes.

LSB Encoding Process

The LSB encoding process involves the following steps:

- Convert secret data: The secret information to be hidden is converted into a binary format.
- Iterate through pixels: Each pixel in the image is accessed sequentially.
- Retrieve pixel values: The RGB color values of each pixel are retrieved.
- Modify LSBs: The least significant bit(s) of the color values are altered to match the corresponding bits of the secret data.
- Update modified pixels: The modified pixel values are updated in the image.

Embedding Capacity and Visual Impact

- The LSB algorithm's embedding capacity depends on the number of LSBs modified per pixel. By modifying a single LSB, the algorithm achieves a higher embedding capacity but may introduce more visual distortion. Conversely, modifying fewer LSBs minimizes visual impact but reduces the amount of data that can be concealed.

LSB Decoding Process

To extract the hidden data from the steganographic image, the LSB decoding process is performed:

- Iterate through pixels: Each pixel in the steganographic image is accessed sequentially.
- Retrieve pixel values: The RGB color values of each pixel are retrieved.
- Extract LSBs: The least significant bit(s) of the color values are extracted to reconstruct the binary representation of the hidden data.

Data Recovery and Verification

Once the binary representation of the hidden data is reconstructed, it can be converted back to its original format (e.g., text, binary, or any other data type). Data recovery also involves verifying the integrity and authenticity of the extracted data to ensure its reliability.

Strengths and Limitations of the LSB Algorithm

The LSB algorithm offers several strengths, including simplicity, ease of implementation, and minimal visual distortion. However, it has some limitations, such as susceptibility to steganalysis techniques that analyze statistical properties of LSB-altered images. Additionally, LSB manipulation may introduce slight visual artifacts, especially if many LSBs are modified.

The LSB algorithm provides a fundamental approach to hide data within digital images for the purpose of secure communication and covert information transfer. While it offers simplicity and moderate security, it should be supplemented with additional techniques, such as encryption and other steganographic methods, to enhance security and withstand advanced detection techniques.

CHPATER 8

RESULTS

The result of image steganography is a steganographic image that appears visually similar to the original cover image but contains hidden data embedded within it. The exact nature of the result depends on the specific steganography technique used and the parameters chosen during the embedding process.

When the steganographic image is viewed without knowledge of the hidden data, it should not raise suspicion or indicate the presence of any additional information. The goal is to make the steganographic modifications undetectable to casual observers.

To extract the hidden data from the steganographic image, one needs to use a compatible decoder or extraction algorithm. The extraction process reverses the embedding process and retrieves the original hidden data, allowing the recipient to access the concealed information.

CHAPTER 9

SNAPSHOTS

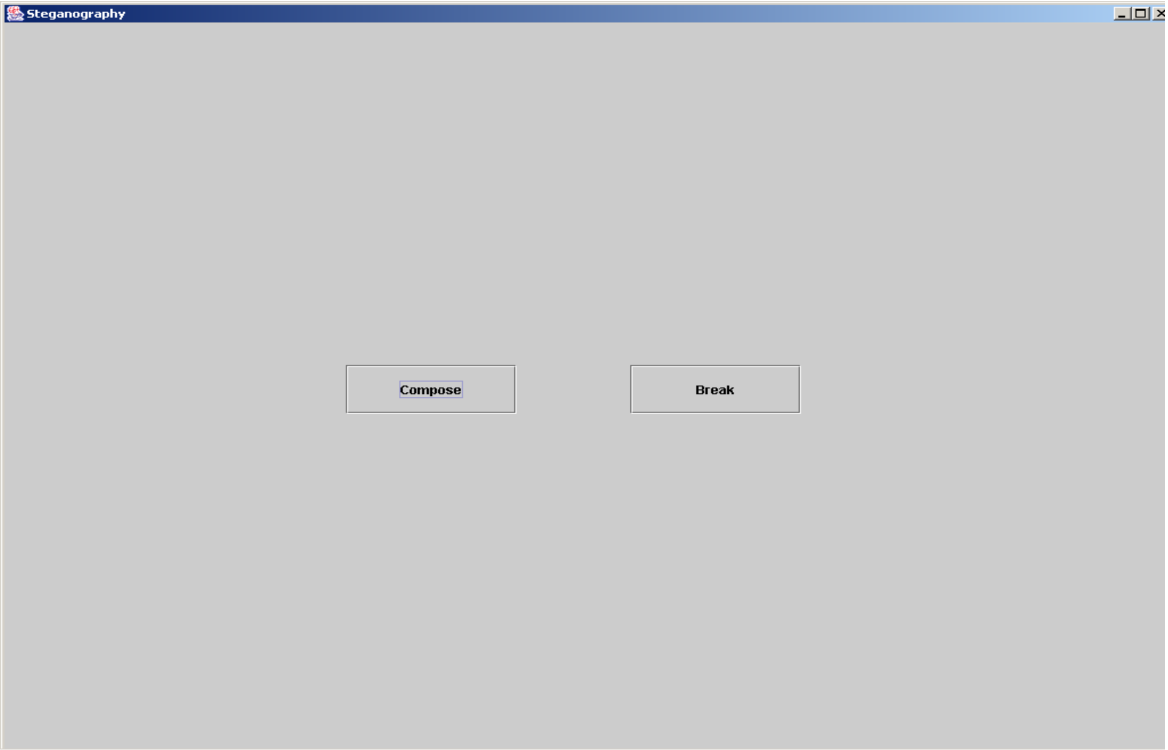
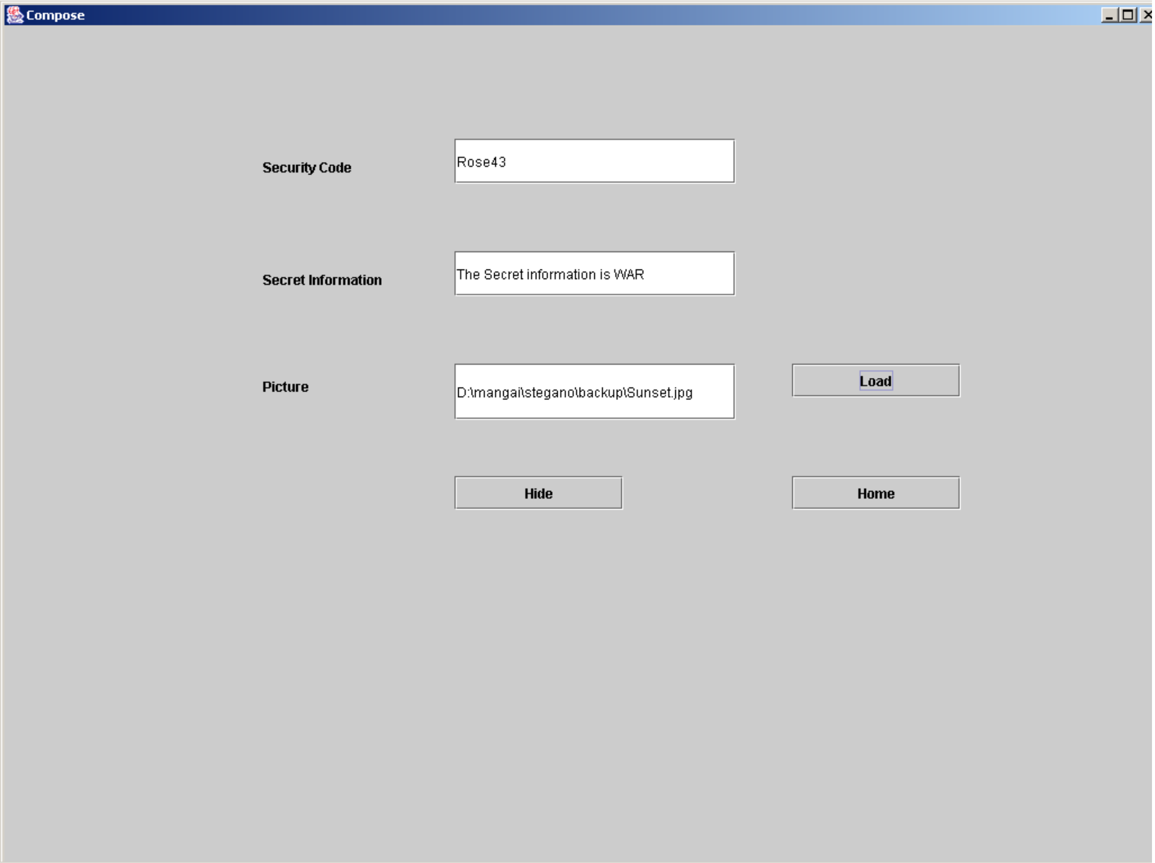


Fig 9.1 : Home Page

In the fig 9.1 when the "Compose" button is clicked, it opens a new window called ComposePage, where users can compose a steganographic message by entering a security code and secret information. The ComposePage window allows users to load an image, hide the secret information within the image, and provides feedback on the process.

Similarly, when the "Break" button is clicked, it opens a new window called BreakPage, which allows users to break/decode a steganographic message hidden within an image. Users can enter a security code and select an image file to extract the hidden information



Security Code: Rose43

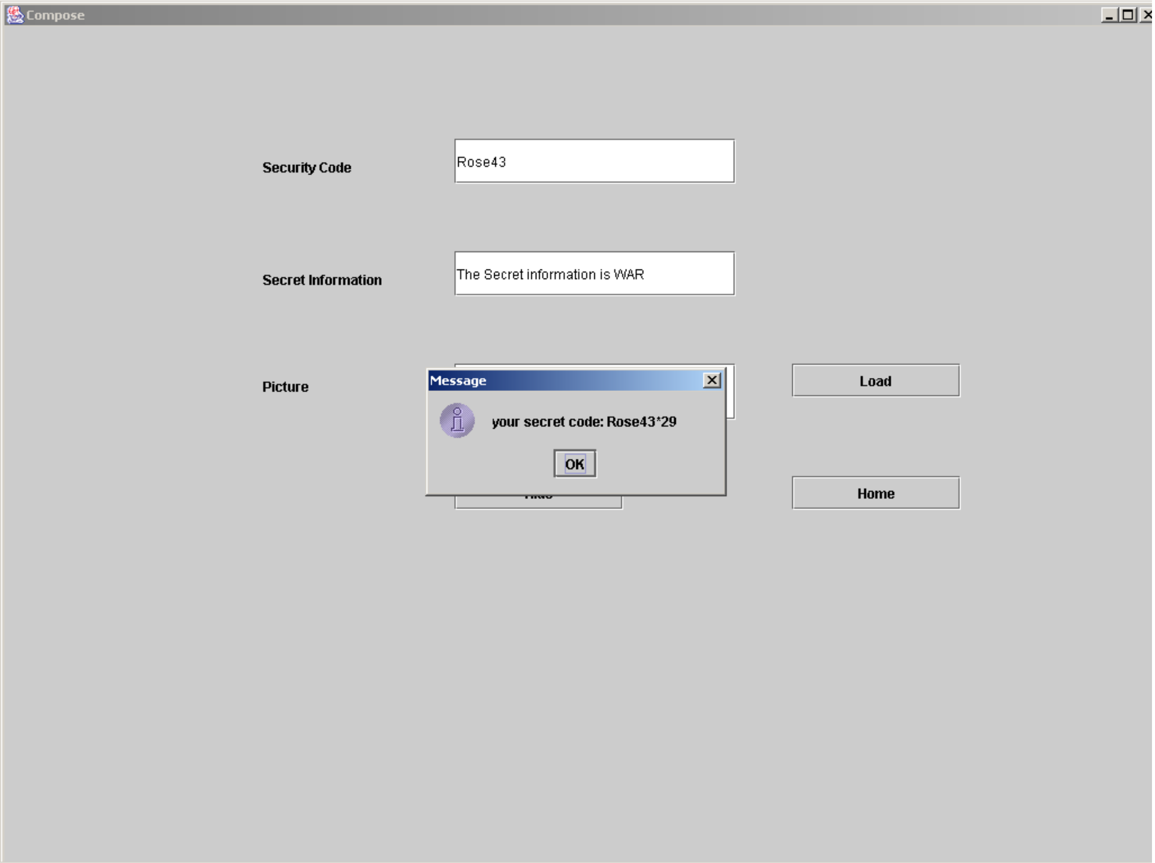
Secret Information: The Secret information is WAR

Picture: D:\mangai\stegano\backup\Sunset.jpg

Fig 9.2 : Compose Page

In the fig 9.2 the GUI window titled "Compose" for the "Compose" functionality of a steganography application.Users can enter a security code, secret information, and the path of an image file.

Clicking the "Load" button opens a file dialog for selecting an image file, and the selected file's path is displayed.Clicking the "Hide" button performs the steganography process: It checks for valid input and extracts the security code, secret information, and image file path.



Security Code Rose43

Secret Information The Secret information is WAR

Picture

your secret code: Rose43*29

Fig 9.3 : Code generation

In the fig 9.3 the code hides the secret information within the image.It generates a new image with the hidden information and displays it in the GUI.A dialog box shows the generated secret code.

The code also creates a thumbnail image of the processed image and saves it as a JPEG file named "secpic.jpg".The "Home" button allows users to return to the main menu.

Overall, the Compose functionality provides an interface for hiding secret information within an image and displaying the result in a GUI window.

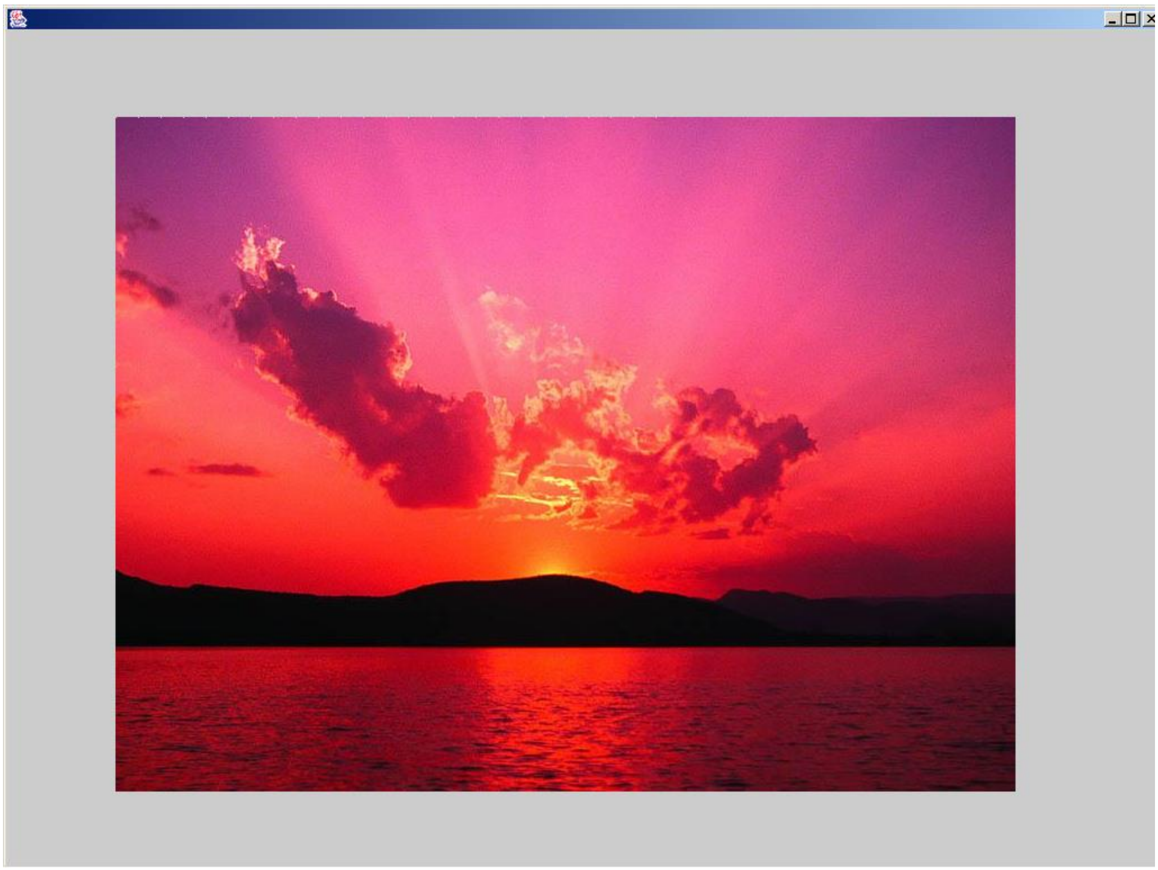
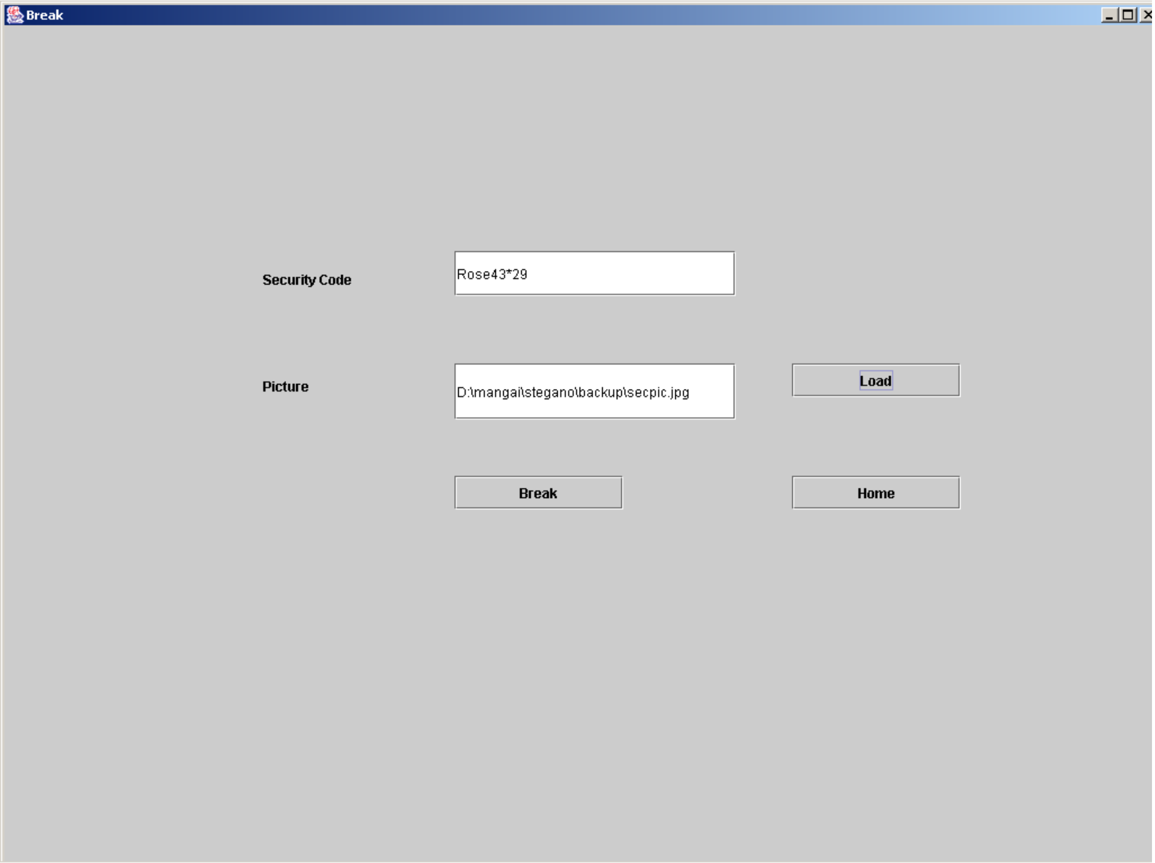


Fig 9.4 : Covering media

In the fig 9.4 image used to hide the secret information. The code supports various image formats, retrieves the dimensions of the loaded image (width and height), and captures the pixel data necessary for further processing, such as hiding the secret information within the image

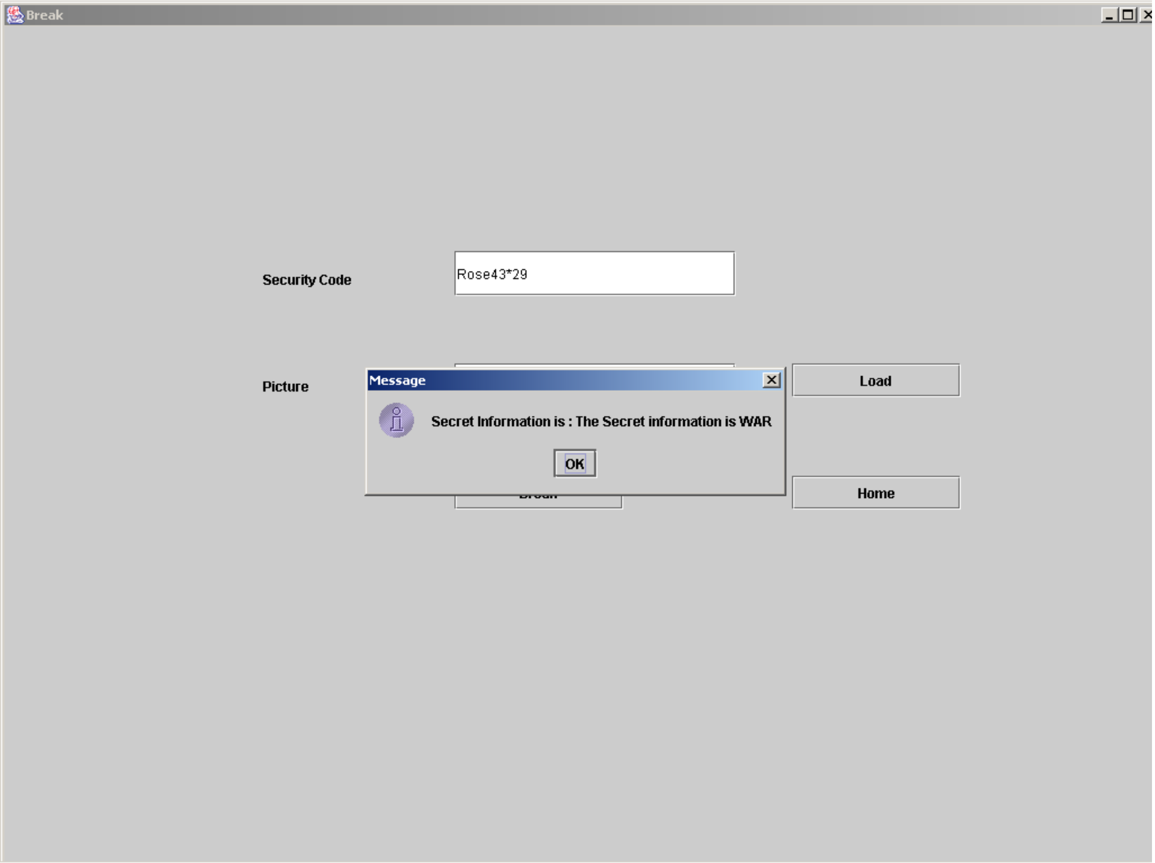


Security Code: Rose43*29

Picture: D:\mangai\stegano\backup\secpic.jpg

Fig 9.5 : Break Page

In the fig 9.5 the "Break" page provides a GUI where users can enter a security code and the file path of an image.Users can load an image file by clicking the "Load" button and selecting the desired file. Clicking the "Break" button initiates the process of extracting hidden information from the loaded image.



Security Code Rose43*29

Message

Secret Information is : The Secret information is WAR

Fig 9.6 : Extracting hidden information

In the fig 9.6 the code analyzes the image pixels, comparing them with the provided security code, and extracts the hidden information. If the security code is valid and matches the hidden information, a message dialog displays the extracted secret information.The "Home" button allows users to navigate back to the main menu or home screen of the application.

CHAPTER 10

CONCLUSION

In conclusion, the provided code represents a basic implementation of a steganography application in Java. The project utilizes Java Swing and AWT libraries to create a graphical user interface and perform image manipulation operations. It allows users to compose a steganographic message by hiding secret information within an image and also provides a functionality to break and extract the hidden message from an image.

While the code serves as a starting point for a steganography application, it should be noted that it is a simplified version and may require further enhancements and considerations to make it a fully functional and secure system.

To expand and improve upon this project, several aspects can be explored:

Advanced Steganography Techniques: Consider implementing more robust and secure steganography algorithms, such as transform domain techniques or hybrid approaches, to enhance the hiding and extraction process.

Encryption and Security: Integrate stronger encryption algorithms and security measures to protect the hidden data from unauthorized access and improve the overall confidentiality and integrity of the system.

User Experience: Enhance the user interface design, usability, and interactivity of the application to provide a seamless and intuitive experience for users.

Error Handling and Validation: Implement comprehensive error handling mechanisms and input validation to handle edge cases, ensure data integrity, and provide meaningful error messages to users.

Performance Optimization: Consider optimizing image processing operations and algorithms to improve the system's performance, especially when dealing with large or high-resolution images.

Overall, the project provides a foundation for building a steganography application but requires further refinement and expansion to create a more robust, secure, and user-friendly system. By incorporating advanced techniques, implementing stronger security measures, and refining the user experience, the project can be developed into a comprehensive and practical steganography solution.

CHAPTER 11

REFERENCES

1. B. Champakamala K. Padmini and D. Radhika "Least significant bit algorithm for image steganography" International Journal of Advanced Computer Technology vol. 3 no. 4 pp. 34-38 2014.

2. V. Tyagi "Image steganography using least significant bit with cryptography" Journal of global research in computer science vol. 3 no. 3 pp. 53-55 2012.

3. C. Varade D. Shaikh G. Gund V. Kumar and S. Qureshi "A technique for data hiding using audio and video steganography" International Journal of advanced Reseach in Computer Science and Software Engineering vol. 6 no. 2 2016.

4. Y. Zhang C. Qin W. Zhang F. Liu and X. Luo "On the faulttolerant performance for a class of robust image steganography" Signal Processing vol. 146 pp. 99-111 2018.

5. J. Fridrich, Steganography in Digital Media: Principles, Algorithms, and Applications. Cambridge, U.K.: Cambridge Univ. Press, 2009.

6. A. S. Panah, R. Van Schyndel, T. Sellis, and E. Bertino, ‘‘On the properties of non-media digital watermarking: A review of state of the art techniques,’’ IEEE Access, vol. 4, pp. 2670–2704, 2016, doi: 10.1109/ACCESS.2016.2570812.

7. M. Hussain, A. W. A. Wahab, Y. I. B. Idris, A. T. S. Ho, and K.-H. Jung, ‘‘Image steganography in spatial domain: A survey,’’ Signal Process., Image Commun., vol. 65, pp. 46–66, Jul. 2018, doi: 10.1016/j.image.2018.03.012.

8. I. J. Kadhim, P. Premaratne, P. J. Vial, and B. Halloran, ‘‘Comprehensive survey of image steganography: Techniques, evaluations, and trends in future research,’’ Neurocomputing, vol. 335, pp. 299–326, Mar. 2019, doi: 10.1016/j.neucom.2018.06.075.