SECURED PALETTE

Safeguarding Secrets with LSB Steganography and Visual Cryptography

JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY

NOIDA

December - 2023

Work Summary

(Minor Project-1)

Enrollment numbers	21103042	21103150	21103160
Names of students	Astha Raghuwanshi	Jahanvi Gupta	Ragini Mittal
Name of supervisor	Ms. Mradula Sharma		

MOTIVATION BEHIND THE PROJECT

In the contemporary landscape of digital communication, the need for secure information sharing has become paramount. In many real-world scenarios, information is often shared among multiple participants, requiring a system that facilitates collaborative decryption. Leveraging the principles of steganography and visual cryptography the project facilitates secure collaboration and information sharing.

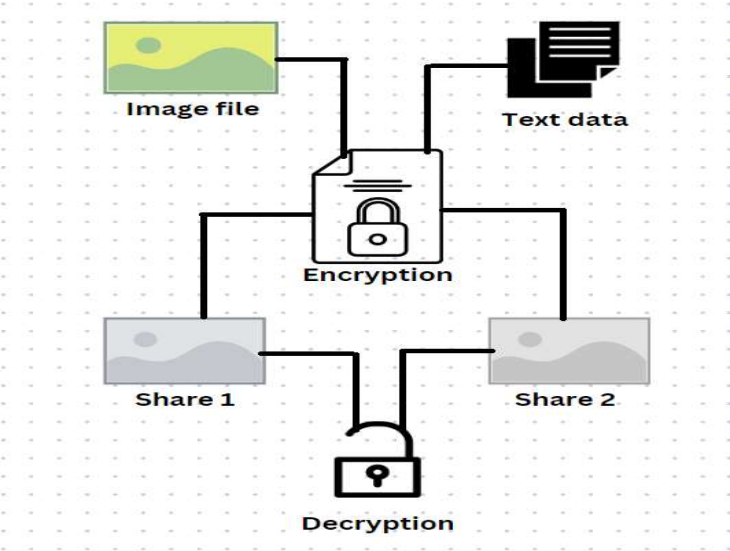
TYPE OF PROJECT

Research cum development project

TECHNOLOGIES USED

1. MATLAB to Python Conversion: The original steganography code was initially developed in MATLAB for concept analysis.
2. Streamlit for Web Development: Our web application is powered by Streamlit, a Python library designed for creating interactive and data-driven web applications
3. Python as the Primary Programming Language: Python emerged as the primary programming language for its versatility, extensive libraries, and widespread community support.
4. Git for Version Control: Git, a distributed version control system, was employed to track changes in the codebase, manage collaborative development, and ensure project stability.

OVERALL DESIGN



FEATURES BUILD

1. Image Selection: Users can choose any image from their local device to serve as the cover for hiding the secret message. A user-friendly interface allows for easy navigation and selection of images.
2. Message Input: The web interface includes a text input field where users can type the secret message they want to embed in the selected image.
3.Embedding Process: Streamlit widgets enable users to initiate the embedding process with a button click.
4. Share Upload: Users can upload the generated shares of the encoded image through the web interface.
5.Decoding Process: The web interface includes interactive controls to initiate the decoding process. Visual feedback informs users about the progress and success of the decoding operation.
6. Message Display: Once the decoding is successful, the secret message is displayed on the interface for the user to read. Clear and concise messaging ensures a straightforward user experience.

METHODOLOGY

1. Requirements: Defining the project objectives, specifying the type of information to be secured, the level of security required, and the collaborative aspects involved.
2. Literature Review: Conducting an in-depth review of existing steganography and visual cryptography techniques, focusing on their strengths, weaknesses, and real-world applications.
3. System Design: Selecting steganography technique: LSB steganography. Designing the visual cryptography scheme. Developing a user-friendly interface using streamlit
4. Implementation: Integrate steganography scheme and visual cryptography algorithm to implement secure communication.
5. Collaborative Decryption: Developing the mechanism that dictates how shares should be overlapped to reveal the hidden message.
6. Documentation: Preparing a comprehensive user manual detailing how participants can use the system for secure information sharing. Documenting the technical details of the implementation, algorithms used, and security measures.

ALGORITHM & DESCRIPTION OF THE WORK

Steganography :

Encoding data in the image:

```
# Putting modified pixels in the new image
newimg.putpixel((x, y), pixel)
if (x == w - 1):
    x = 0
    y += 1
else:
    x += 1
```

Decoding data from the image:

```
# string of binary data
binstr = ''

for i in pixels[:8]:
    if (i % 2 == 0):
        binstr += '0'
    else:
        binstr += '1'

data += chr(int(binstr, 2))
if (pixels[-1] % 2 != 0):
    return data
```

Visual Cryptography:

Generating shares:

```
# Split image based on random factor
n = int(np.random.randint(data[i, j, k] + 1))
img1[i, j, k] = n
img2[i, j, k] = data[i, j, k] - n
```

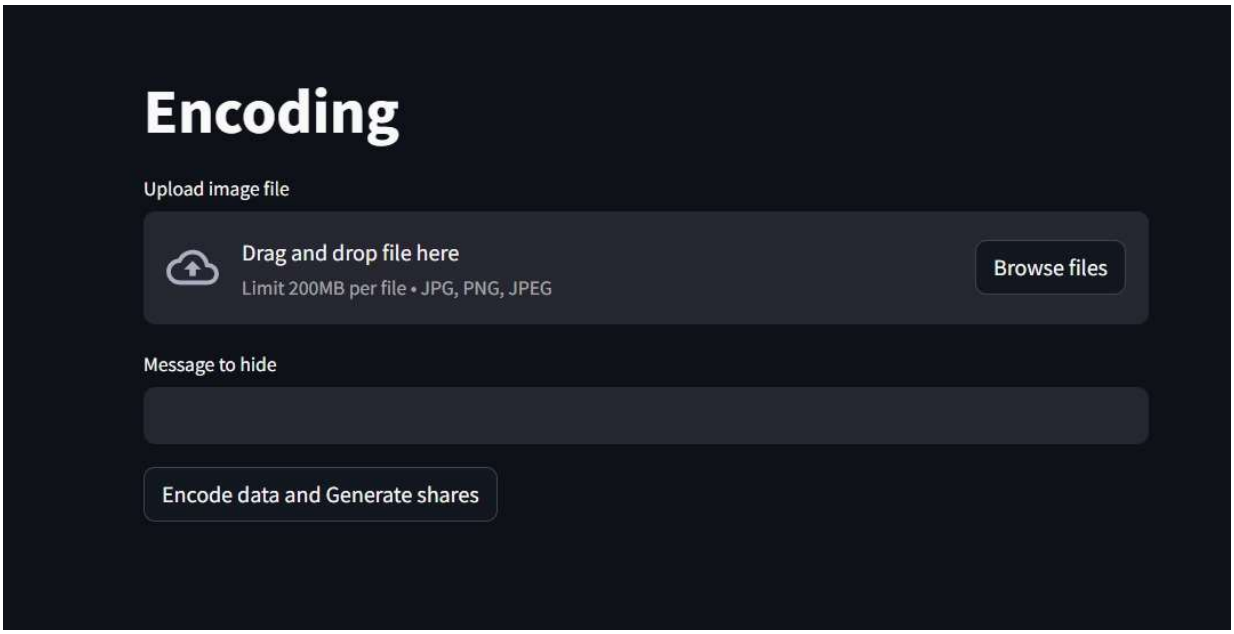
Compressing shares:

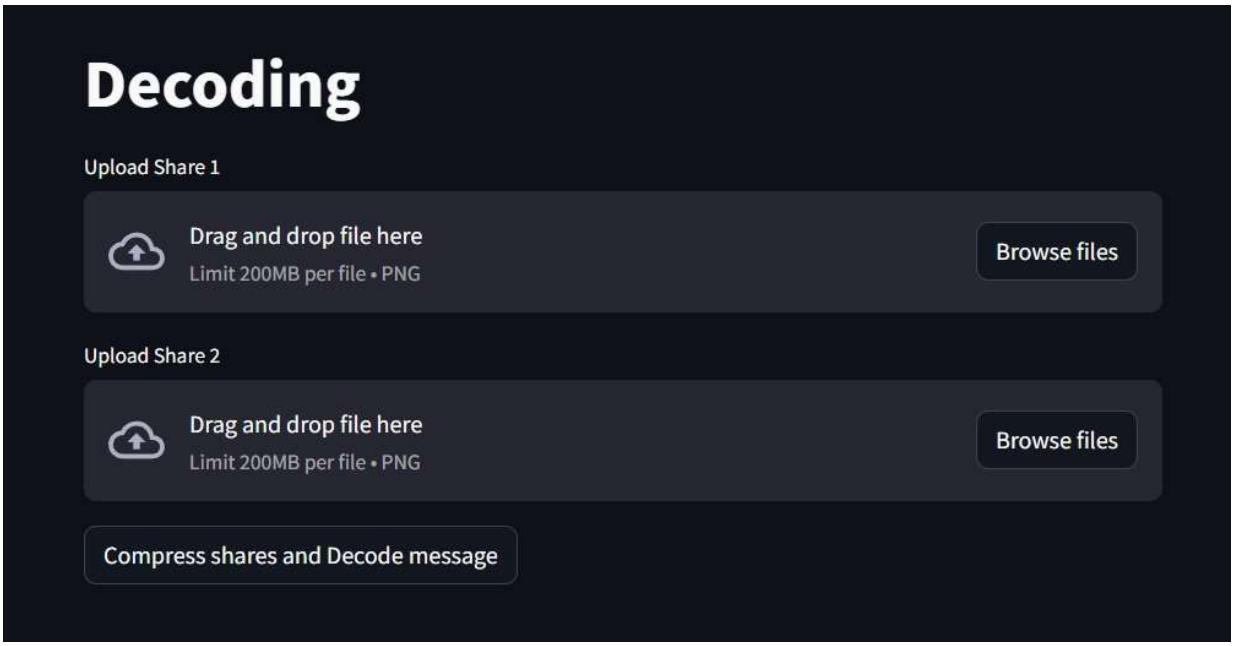
```
img[i, j, k] = img1[i, j, k] + img2[i, j, k]
```

DIVISION OF WORK

Visual Cryptography implementation: Astha Raghuwanshi
Steganography integration: Ragini Mittal
Streamlit Web Interfacing: Jahanvi Gupta

RESULTS





CONCLUSION

In conclusion, our project successfully integrates steganography and visual cryptography to achieve secure multi-user information sharing. By mitigating traditional encryption limitations, we've developed a practical solution that balances confidentiality and collaboration. Through systematic design and testing, the project addresses key challenges in digital communication, providing a pragmatic approach to secure data exchange.