IMAGE STEGANOGRAPHY SYSTEM

Abstract

This project aims to advance image steganography in cybersecurity by leveraging Python's extensive libraries and computational power to develop robust, efficient, and user-friendly methods for embedding and extracting hidden data within digital images. By utilizing Python's capabilities, such as advanced image processing libraries (Pillow, OpenCV) and machine learning frameworks (TensorFlow, PyTorch), the project will focus on creating sophisticated steganographic algorithms that enhance data security while minimizing detectable alterations to image quality. The goal is to improve the resilience of steganographic techniques against common attacks and image manipulations, optimize performance for large-scale applications, and provide intuitive tools for both developers and end-users. Through this approach, the project seeks to strengthen data protection strategies in an increasingly digital and interconnected world.

Languages Used

- Python
- Libraries and Frameworks(Pillow,open cv,NumPy,SciPy)
- Machine Learning Frameworks(Tensor Flow,PyTorch)
- Cryptographic Libraries(PyCryptodome, cryptography)
- GUI and Visualization Tools(Tkinter, PyQt, Matplotlib)
- Web Technologies(Flask)

Input

- Image Inputs (eg:primary image ,additional images)
- Data to Embed (eg: data to hide)
- Steganographic Parameters (using different techniques to hide data..)
- Image Processing Parameters (eg:JPEG,PNG)
- User Interface Inputs(if GUI is developed)
- Output Specifications
- Detection and Analysis Tools(to test the robustness)
- Error Handling and Logging(to track issues)

Output

- Stego Image having hidden data
- Extracted Data could be any message
- Metadata useful for debugging
- Error and Status Reports
- Quality and Integrity Analysis
- User Interface Outputs gives visual feedback
- Cryptographic Outputs shows both encrypted and decrypted results
- Operation Reports and Documentation