TRIBHUWAN UNIVERSITY
INSTITUTE OF ENGINEERING

Khwopa College of Engineering
Libali, Bhaktapur
Department of Computer Engineering

A Proposal
on
Image Steganalysis Using Ensemble Classifiers

Submitted in partial fulfillment of the requirements for the degree

BACHELOR OF COMPUTER ENGINEERING

Submitted by:

Sachin Koirala	KCE077BCT029
Sajal Poudel	KCE077BCT031
Unique Shrestha	KCE077BCT045
Utsav Chandra Kayastha	KCE077BCT046

Khwopa College of Engineering
Libali, Bhaktapur
09 December 2023

Abstract

Steganography is a covert visual attack, that involves hiding malicious data inside innocent-looking carrier information. It's a technique for hiding information within an image, audio, or video file in such a way that the hidden information is not readily apparent to the human eye or ear. Digital images are the most common carrier format for steganography due to their frequent use on social media, websites, and email. Almost two-thirds of internet is made up of JPEGs, which serve as perfect carriers for these types of malware. Hence, it is important to have strong steganalysis methods. Our paper focuses on using ensemble classifiers as well as Generative Adversarial Networks (GAN) to detect hidden malicious contents in carrier files. Ensemble classifiers is made up of various models working independently, employed to identify images modified using various steganography algorithms. These models are integrated into another model which utilizes algorithms such as logistic regression, to detect the presence or absence of malicious data. Generative Adversarial Networks (GAN) plays a crucial role in generating steganographically modified images, serving as a testing ground to ensure the effectiveness of the ensemble classifier models in successfully detecting and analyzing potential threats of steganogrpahically modified carriers. The core objective is to enhance steganalysis accuracy by integrating Machine Learning algorithms to counter the ever evolving field of steganography.

Table of Contents

List of Figures

Chapter 1

Introduction

1.1 Background

Steganography is like hiding a secret message, like a picture or music. It's a way of keeping your message private by making it blend in, so others don't even realize there's a secret there. It has many types such as : image steganography, audio steganography, video steganography.

Image Steganography

Image Steganography is the process of hiding information which can be text, image or video inside a cover image. The secret information is hidden in a way that is not visible to the human eyes. Different techniques of image steganography:

- **Transform Domain Steganography**
  These techniques involve transforming the image data into a different domain (e.g., frequency domain using Discrete Cosine Transform or wavelet domain using Discrete Wavelet Transform) and then hiding information in the transformed coefficients.

- **Compressed Domain Steganography**
  Hides information within the compressed data of an image file to reduce file size and detection difficulty.

- **Least Significant Bit (LSB) Technique**
  It involves replacing the least significant bits of the pixel values with the secret message bits.

- **Pixel Value Differencing (PVD) Technique**
  Identifies and modifies pixels with small value differences to encode information in both grayscale and color images.

1.2 Problem Statement

The development of image steganography poses a serious cybersecurity threat. Since images play a major role in digital communication, the use of advanced steganographic techniques to hide malicious data inside another carrier information pose an intricate threat to the security of a system. Creative approaches are required since, secretly implanted malicious programs are difficult for traditional steganalysis to accurately identify. The complexity of compression and encryption methods adds an additional level of difficulty in detection. Existing steganalysis which were created for traditional steganography, are not flexible enough to detect the ever evolving algorithms for steganography. Thus, in order to protect the integrity of digital communication. This proposal seeks to propose an ever evolving steganalysis process created with the help of ML to detect and tackle the subtle changes caused by concealing of malicious data within unsuspecting carrier files.

1.3 Objectives

- **Adaptability to Emerging Techniques:** Develop steganalysis models that can adapt to emerging steganographic methods by continuously learning and updating their knowledge base.
- **Algorithmic Innovation:** Explore novel algorithms and methodologies tailored to analyze the intricate patterns and structures associated with image-in-image steganography.
- **Enhanced Feature Extraction**: Investigate and optimize feature extraction techniques to capture subtle anomalies indicative of image-in-image concealment, ensuring high detection accuracy.
- **Adaptability to Diverse Image Formats:**Develop steganalysis models capable of detecting concealed images across a variety of image formats, resolutions, and compression methods.
- **Benchmarking and Evaluation:** Establish a comprehensive benchmark dataset containing images with various steganographic content for testing and evaluating the performance of developed steganalysis techniques.
- **Development of Specialized Steganalysis, Techniques:** Create and apply steganalysis methods with a particular goal in mind: identifying images that are hidden inside of another.
- **Algorithmic Innovation:** Investigate modern techniques and algorithms designed specifically to examine the complex structures and patterns connected to image-in-image steganography. Flexibility to Diverse Image Formats: Create steganalysis models that can identify hidden images in a range of image formats, compression techniques, and resolutions.

- **Real-time Implementation:** Explore the integration of the developed models into real-time steganalysis systems. Evaluate their performance in dynamic environments.
- **Benchmarking and Evaluation:** To test and evaluate the effectiveness of created steganalysis tools with photos that have different steganographic content.

Chapter 2

Literature Review

Chapter 3

Feasibility study

Chapter 4

Methodology

4.1 Software Development Approach

Agile is an iterative process-based approach to software development. In the Agile process model, work is broken down into more manageable, smaller iterations without requiring a lot of long-term planning. The requirements and scope of the project are determined early on, and the number, length, and scope of each iteration are preplanned. Each iteration is considered as a short time ”frame” in the Agile process model, which lasts for a few weeks. In each iteration, teams move through the phases of the software development life cycle, which include planning, requirements analysis, design, coding, testing, and demonstration of a working product for client review. Agile places a significant value on flexibility, teamwork, and regular client feedback.
The main reason for which we choose this development process:

1. Very quick,flexible and efficient.
2. Risk minimization.
3. Projects are split into sprints for better management and productivity.
4. Through iterative testing and sprints, the final product contains less bugs.
5. Development period for application is reduced.

4.2 Block diagram of proposed system

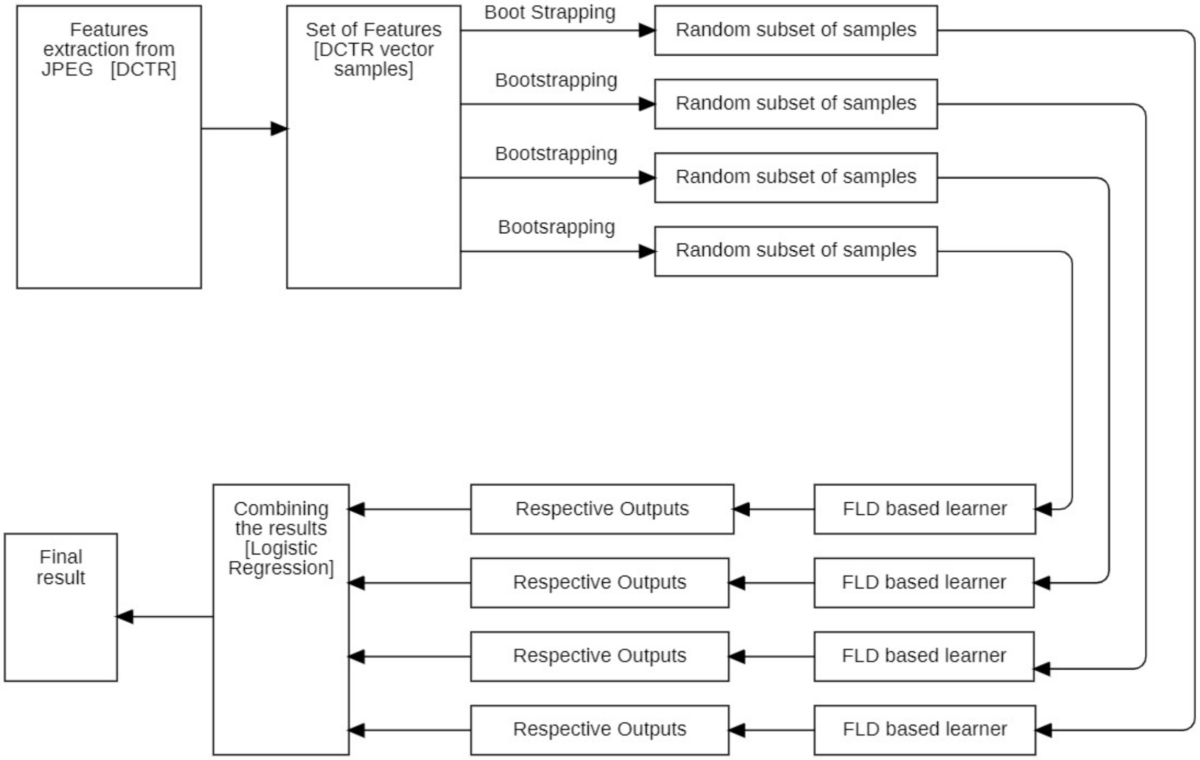

Figure 4.1: Block diagram of proposed system

4.3 Description of working flow of proposed system

this is data flow

Chapter 5

Implementation Plan

5.1 Gantt Chart

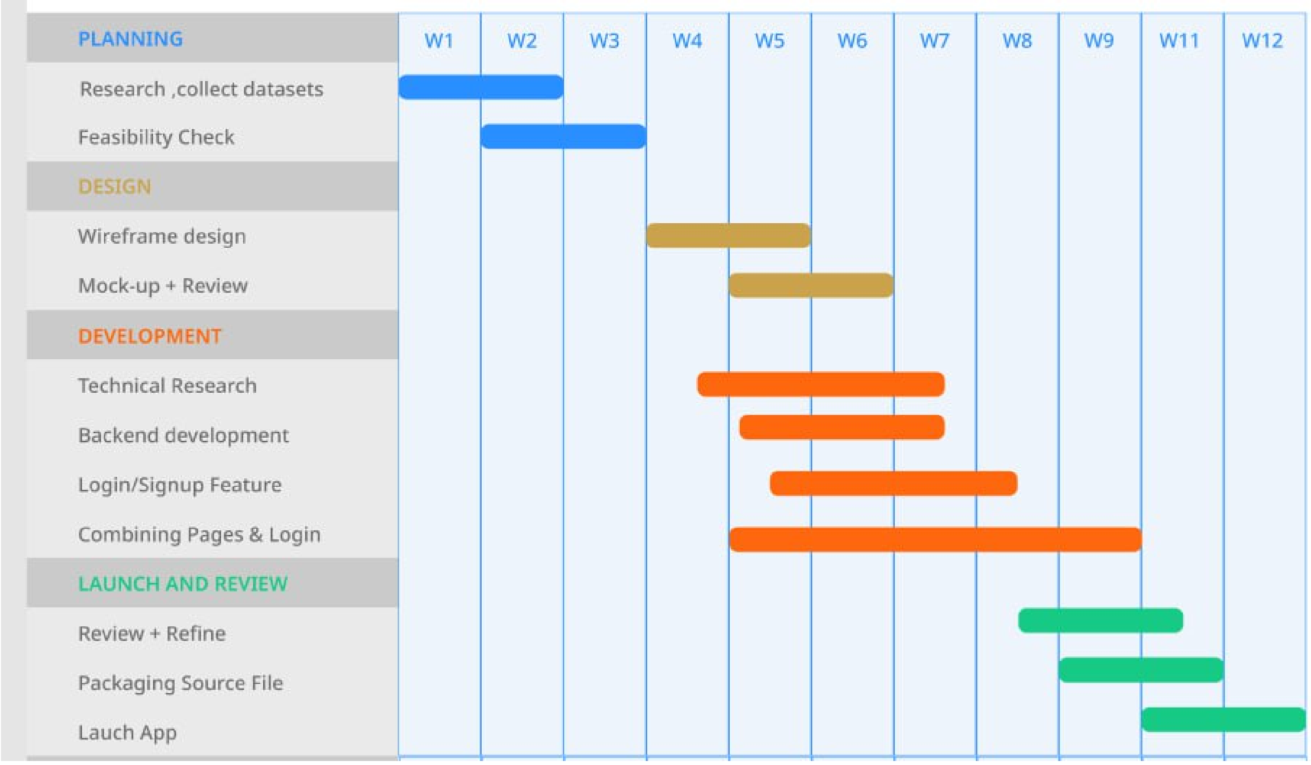

Figure 5.1: Gantt Chart

5.2 Software Requirement

- **python** Python is a versatile programming language commonly used for developing software applications. It can be used for various tasks in the system, such as backend development, data processing, and machine learning integration.

- **MySQL** MySQL is a popular relational database management system (RDBMS) that provides efficient storage and retrieval of data. It can be used to store parking-related information, such as user details, parking space availability, and billing records.
- **React** React is a JavaScript library for building user interfaces, particularly in single-page applications. Developed by Facebook, it uses a declarative approach for efficiently updating the DOM. With a component-based structure, React enhances modularity and reusability, making it a popular choice for creating interactive and scalable web applications.
- **Javascript** JavaScript is a programming language commonly used for developing web-based applications. It can be used for front-end development, implementing interactive features on the system's web interface, and facilitating communication with the backend.
- **Tensorflow** TensorFlow is an open-source machine-learning framework that provides a wide range of tools and libraries for building and deploying machine-learning models. It can be used for image recognition, object detection, and prediction algorithms in the Smart Parking Management System.
- **Keras** Keras is a high-level neural networks API written in Python. It can be used as a user-friendly interface to TensorFlow, simplifying the process of designing and training deep learning models for tasks like number plate recognition or image analysis in the system.
- **VS Code** VS Code is a popular and widely used source code editor that offers a range of features and extensions to enhance the development experience. It supports multiple programming languages, including Python, JavaScript, and React, making it suitable for working with the different components of the system.

5.3 Cost Estimation

Chapter 6

Expected Outcomes

1. ML-based Steganalysis Models: Development of advanced ML models capable of detecting hidden information with high accuracy.
2. Optimized Feature Extraction Techniques: Identification and optimization of feature extration techniques to improve the sensitivity and specificity of steganalysis.
3. Adaptability and Real-time Detection: Models that can adpat to emerging steganography techniques and provide real-time detection capabilities.
4. Robustness Against Adversarial Attacks: Implementation of countermeasures to enhance the models' robustness against adversarial attempts to evade detection. (need to edit after reading research papers)

Chapter 7

References

i am mine [2]. ramdom [1]

Chapter 8

Bibliography

[1] Jeppe Nicolaisen. Citation analysis. *Annual review of information science and technology*, 41(1):609–641, 2007.

[2] EDDIE VEDDER. I am mine, 2002.