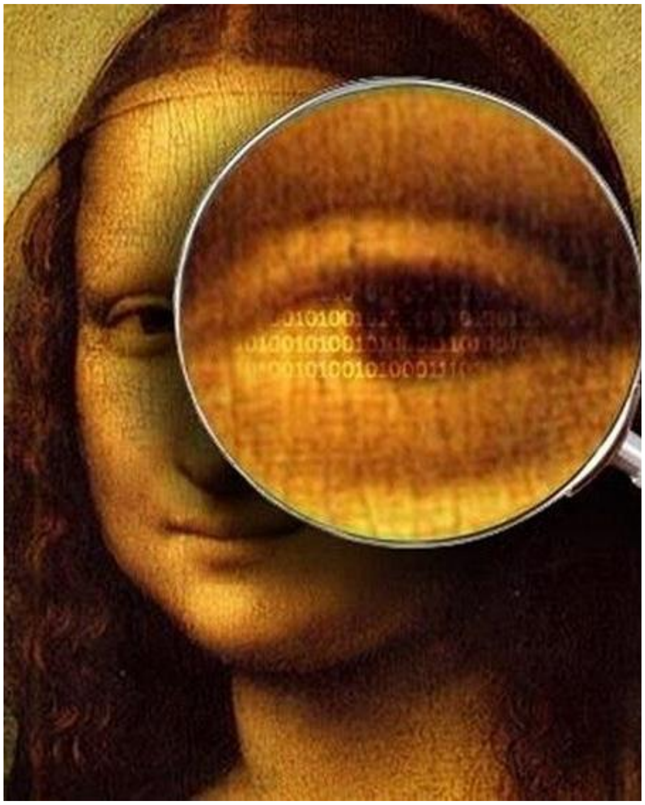

STEGANALYSIS FOR MANAGE ENGINE

Project Proposal Report

DEPARTMENT OF INFORMATION TECHNOLIGY

Tharindu Gihan Indrajith
Cyber Security

Steganalysis for Data Loss Prevention

RP24_25J_003

Project Proposal Report

Tharindu Gihan Indrajith

B.Sc. (Hons) Degree in Information Technology specialized in Cyber Security

Department of Information Technology

Sri Lanka Institute of Information Technology

June 2024

Declaration

I declare that this is my own work, and this proposal does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Group member name	Student ID	Signature
Indrajith G.B.T.G	IT21229220	

The above candidate is carrying out research for the undergraduate Dissertation under supervision of the undersigned.

………………………………..   …………………………….

Signature of the supervisor   Date

(Mr. Amila Senerathna)

………………………………..   ……………………………...

Signature of Co-supervisor   Date

(Ms. Suranjini Silva)

Acknowledgement

I extend my sincere gratitude to my supervisor, Mr. Amila Senerathna, and co-supervisor, Ms. Suranjini Silva, for their invaluable guidance and support throughout this research study. I'm thankful to industry experts for sharing their expertise which helped me get a certain domain knowledge. Special thanks to my team members for their contributions, and to those who aided me willingly. Lastly, my heartfelt appreciation to my family for their constant love, assistance, and encouragement.

Abstract

In the digital age, the security of sensitive information within organizations is paramount. Data exfiltration through covert channels, such as steganography, poses a significant threat to data loss prevention (DLP) efforts. This research proposes the development of a steganalysis tool specifically designed to detect, block, and analyze stego images within an organizational network. The tool will be capable of identifying images that contain hidden data, preventing their transmission to unauthorized recipients, and alerting the relevant authorities. Furthermore, the tool will include functionality to extract the concealed information, allowing for a thorough investigation of potential data breaches. By integrating advanced steganalysis techniques with real-time monitoring and alert systems, this research aims to enhance the effectiveness of DLP strategies, safeguarding sensitive information from unauthorized disclosure.

Keywords : steganalysis, Data Loss Prevention, Covert Channels, Information Security, Digital Forensics, Steganography Detection, Hidden Data Extraction, Network Security, Cybersecurity, Real-time Monitoring

Table of Contents

Contents

List of Figures

List of Abbreviations

Abbreviation	Description
DLP	Data Loss Prevention
SDLC	Software Development Life Cycle
PII	Personally Identifiable Information
DCT	Discrete Cosine Transform
DWT	Discrete Wavelet Transform
JEPG	Joint Photographic Experts Group
PNG	Portable Network Graphic
PGM	Portable Gray Map
CNN	Convolutional Neural networks
PSNR	peak signal-to-noise ratio

1. Introduction

In today's interconnected digital landscape, organizations face an escalating risk of data breaches, with malicious insiders and external attackers continuously seeking innovative ways to circumvent traditional security measures [1]. Among the various methods employed for covert data exfiltration, steganography has emerged as a particularly insidious technique. Steganography allows sensitive information to be hidden within seemingly innocuous files, such as images, making detection and prevention highly challenging. This poses a significant threat to Data Loss Prevention (DLP) strategies, which are essential for protecting an organization's critical assets and maintaining regulatory compliance [2].

As steganographic techniques become more sophisticated, the need for robust detection and analysis tools has never been greater. This research proposes the development of an advanced steganalysis tool aimed at addressing this critical gap in cybersecurity defenses. The tool will be designed to identify stego images within an organization's network, prevent their unauthorized transmission, and alert security teams to potential data breaches [3]. Additionally, it will provide the capability to extract hidden information, enabling a detailed examination of the data being exfiltrated and the intent behind it.

By integrating this tool into newly created DLP framework, organizations can significantly enhance their ability to detect and respond to covert data exfiltration attempts, ultimately safeguarding their sensitive information from unauthorized disclosure. This research aims to contribute to the field of cybersecurity by providing a practical solution to the growing challenge of steganography-based data loss [4].

2. Background and literature survey

Steganography and steganalysis

Steganography is the practice of hiding data within other non-suspicious data to avoid detection. Unlike cryptography, which obscures of a message, steganography conceals the very existence of the message [5].

There are two methods of hiding data in images these are Least Significant Bit (LSB) manipulation and Transform Domain Techniques. The LSB method involves altering the least significant bits of pixel vales in an image. Since changes in these bits are minimal, they are often imperceptible to human eye.

In Transform-Domain Techniques there are two techniques we can follow, these are Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT). DCT is commonly used in JPEG compression. Data is hidden by modifying the DCT coefficients of the image. This method is more robust against image processing operations like compression and cropping. DWT transforms the image into different frequency components. Data can be hidden in the wavelet coefficients, providing a balance between imperceptibility and robustness. Below flowchart represents the steganography process.

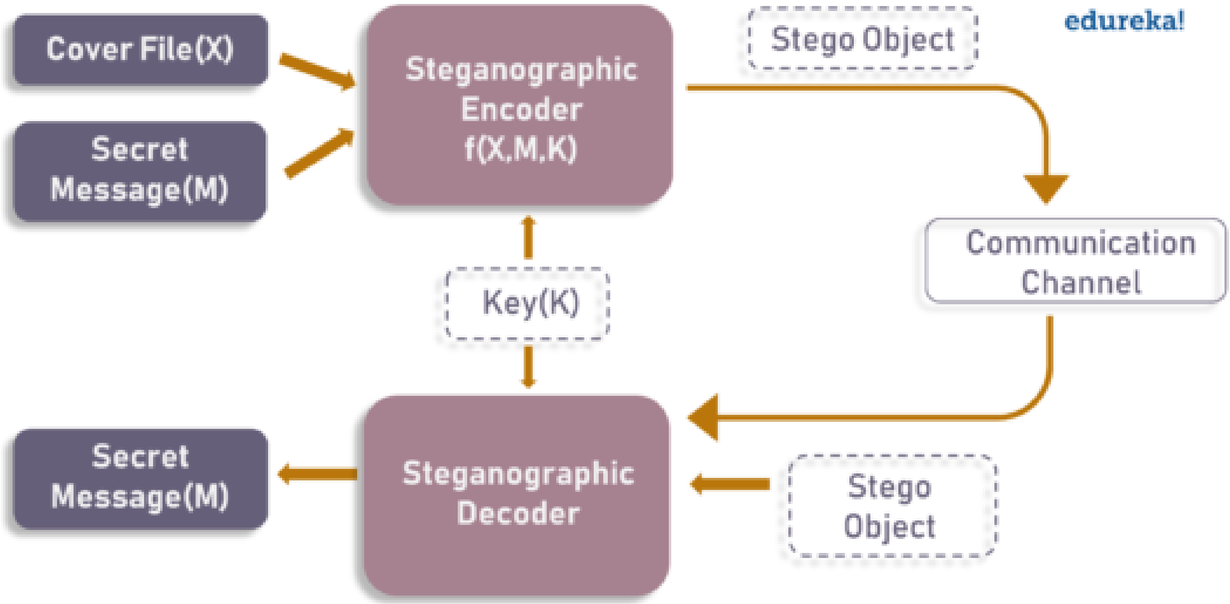

Figure 1 - steganography process.

Steganalysis is the technique of finding out and extracting the data hidden in multimedia files, such as images, audio, and videos. This area, hence, is very crucial in cybersecurity and digital forensics since it reveals the hidden communications that can be conducted for malicious purposes. Steganalysis can prevent the misuse of digital media for unlawful activities by detecting the hidden data.

The more classical steganalysis approaches make use of statistical analysis. For instance, histogram analysis checks the distribution of values of pixels inside an image to decide upon any abnormality. In this respect, hidden information will modify this particular distribution, and after modification, the histogram will look a bit awkward. Another approach, the chi-square attack, checks the uniformity of pixel values. If data is hidden, uniformity will be disrupted, and consequently it gives proof of the presence of steganography.

Pixel-based methods are also among the common approaches in steganalysis. The most straightforward type of steganography is based on changing the least significant bits of the pixel values. Steganalysis may reveal such minor changes by comparing the LSBs of the image with typical patterns. RS analysis is another pixel-based method that considers regular and singular groups of pixels. Changes in those groups may attest to hidden data.

Transformation domain techniques give another dimension to the analysis. The most predominant one is the DCT analysis of JPEG images, which works in analyzing DCT coefficients; this may be modified by hidden data and hence can be detected. Similarly, wavelet transform analysis uses wavelet transforms in order to detect frequency-domain anomalies in an image. These traditional techniques form a basis of different steganalysis techniques and are normally jointly used for the enhancement of detection accuracy. Below flowchart represents the process of steganalysis.

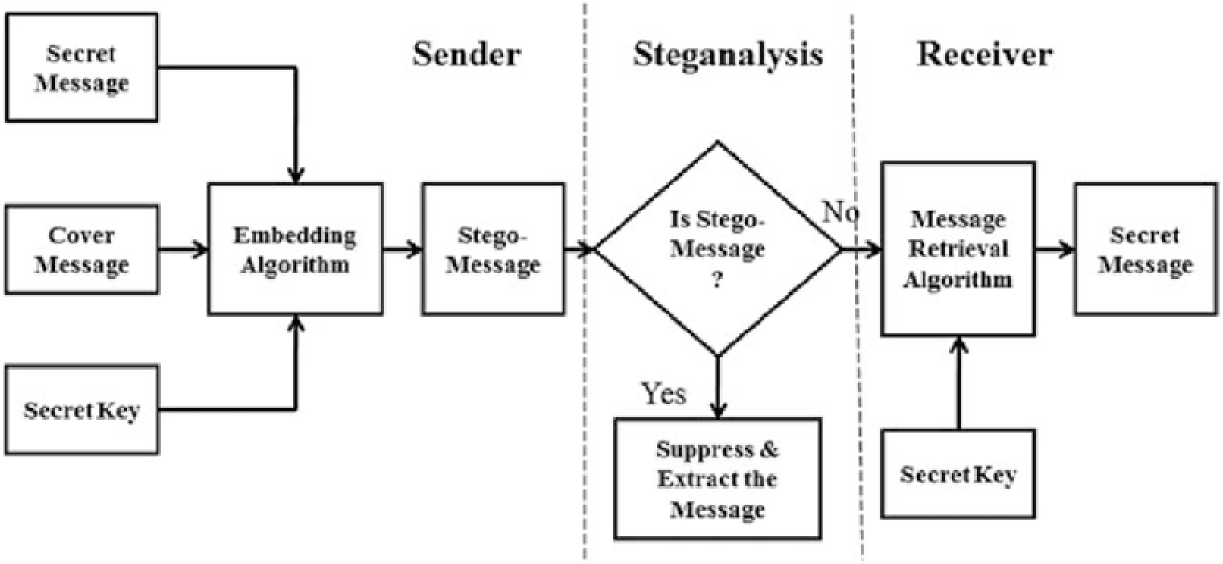

Figure 2 - steganalysis process

Deep learning in steganalysis

On its part, deep learning greatly revolutionized steganalysis by offering progressive means through which hidden data within files in multimedia can be detected. The introduction of CNNs and other neural architectures raised the capabilities for handling and analyzing images to levels where subtler patterns and anomalies, which usually would have easily escaped traditional methods, could be picked out. Below flowcharts represents the difference between machine learning based steganalysis and deep learning based steganalysis.

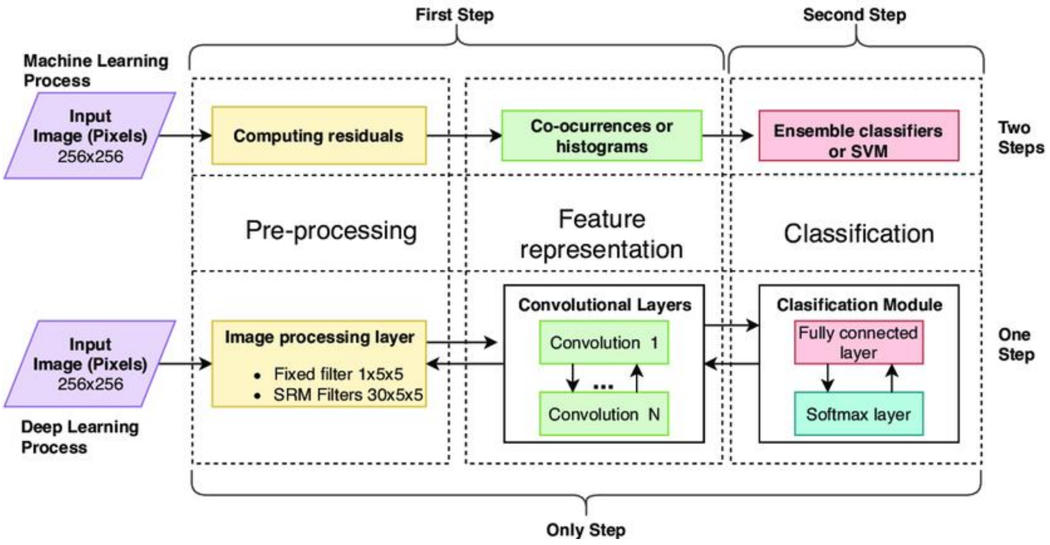

Figure 3 - difference between machine learning and deep learning steganalysis

The most critical advantage of deep learning-based approaches is the ability of finding complex patterns in large data. Other steganalysis approaches either based on statistic analysis or pixel-based approaches depend upon rules predefined by human beings, and they do not work when applied steganographic methods are complicated or sophisticated. While in deep learning methods, one can learn from big data and its precision can be improved through several iterations, which makes these models robust. For example, CNNs perform especially well on image processing tasks due to the hierarchical nature of their spatial features captured across many different levels of abstraction [6].

Despite these advantages, there are several challenges associated with deep learning-based steganography detection. The major ones relate to very large and diverse training datasets. In essence, substantial labeled data is required, which may be hard to gain, especially for some rare or novel steganographic techniques. Besides, while high accuracy can be achieved by deep learning models, they may still struggle to detect highly sophisticated or adaptive steganographic methods that evolve over time [7].

While deep learning has indeed brought unparalleled advancement to steganalysis, the challenges discussed here call for ongoing research and development that promise to take such advanced models even further.

Data Loss Prevention (DLP) tools

Data Loss Prevention (DLP) tools are designed to protect sensitive data from being lost, leaked, or accessed by unauthorized users. These tools monitor, detect, and prevent data breaches by enforcing security policies across an organization’s data in use, data in motion, and data at rest [8]. They can include features like monitoring user activities, blocking unauthorized data transfers, and encrypting sensitive information [9].

In the context of DLP tools, steganalysis cam be used to uncover covert data exfiltration attempts where sensitive information is hidden seemingly innocuous files [10]. By integrating steganalysis techniques, DLP tools can enhance their ability to detect and prevent sophisticated data leakage methods that might otherwise go unnoticed [11].

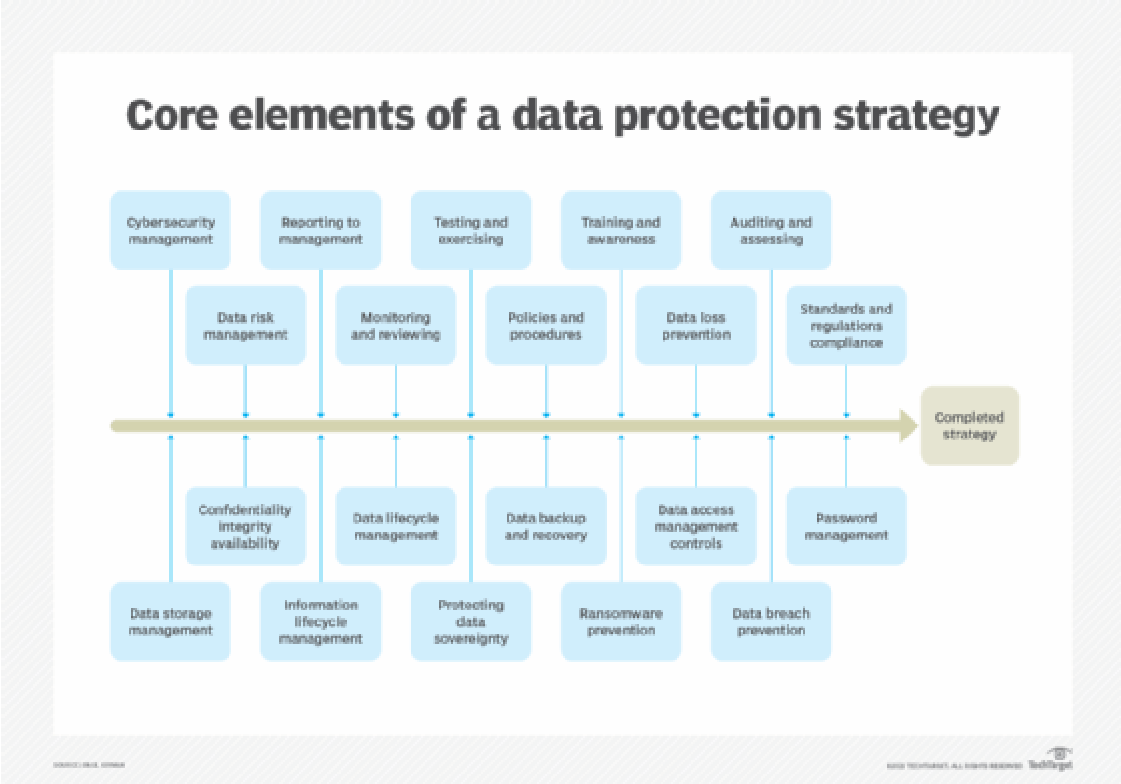

Figure 4 - core elements of a data protection strategy

Related work

One approach involves destroying stego images formed by adaptive embedding methods. Progonov (2023) evaluates dictionary learning methods for stego image destruction, focusing on minimizing changes in image statistics to avoid detection by attackers, making this a preventive measure in DLP systems [12].

Another study by Bhuva et al. (2021) emphasizes the effectiveness of StegoAppDB and StegHide tools, analyzing the data hiding efficiency of various steganographic algorithms and their implications for DLP by using metrics like PSNR and embedding techniques [13].

Ahmad (2023) proposes a fuzzy logic and CNN-based steganalysis framework for detecting hidden data in adaptive steganography, highlighting its utility in pinpointing hidden data for policy enforcement in DLP systems [14].

Moreover, Monika and Eswari (2022) propose a stegware neutralization model, which efficiently detects and neutralizes stego-malware in cloud environments, offering a robust mechanism for preventing data loss [15].

Finally, Hidayasari et al. (2020) introduce CNN Yedrodj-net, which effectively performs blind steganalysis, an essential capability for DLP tools to detect unknown steganographic methods [16].

3. Research Gap

As discussed in the above literature review, a notable gap can be seen in the existing systems. Below mentioned are the research gaps found.

Application Reference	Multi-scale CNN utilization	Real-Time detection	Feature fusion	DLP integration	Computational efficiency	Steganalysis in diverse formats
Reference [17]	✘	✓	✘	✘	✓	✓
Reference [18]	✘	✓	✓	✘	✓	✓
Reference [19]	✘	✘	✘	✘	✓	✘
Reference [20]	✘	✓	✓	✘	✓	✓
Proposed system	✓	✓	✓	✓	✓	✓

Complementary to the existing research in this area, this solution offers a fresh approach with two different CNN models working on two different scales-fine and coarse-for the detection of steganalysis, which allows analysts to detect hidden threats with high accuracy and to precisely locate the exact position by narrowing down the detection to specific image patterns across multiple scales. After extracting these features at multiple scales, a feature fusion mechanism will be applied in order to combine the outputs of both CNN models, hence coming up with a more robust representation against possible steganographic content.

The detection will be performed by the system in real-time, hence being suitable for installations in DLP solutions. It also allows for a module for ranking the quality of the detection, which enables the elimination of false positives and increases the reliability of the detection. Threat categorization will also be possible, making it easier for users to filter the steganalysis results according to file formats or data type.

After the detection and categorization are done, the results will be presented through an intuitive web interface, easily integrable with any existing DLP system. The interface will further support automated workflows with customized notifications, thus efficiently managing and executing threat responses. In general, this solution will contribute to enhancing data security by embedding advanced steganalysis in DLP tools, improving not only the detection accuracy but also operational efficiency.

4. Research Problem

The key research problems being faced by steganalysis implementations in Data Loss Prevention (DLP) tools are as follows:

- **Feature extraction challenges:** Single-scale CNN models often struggle to capture steganographic patterns across different levels of image granularity. This leads to the loss of important features that could help in detecting more sophisticated steganography methods. The challenge lies in extracting meaningful features at multiple scales to improve detection robustness.
- **Real-Time detection:** Implementing CNN models with high detection accuracy is often computationally expensive, making real-time detection in DLP environments difficult. Existing solutions frequently suffer from high processing times, hindering the tool's ability to detect and respond to steganographic threats promptly.
- **False Positives and Accuracy:** Current steganalysis techniques tend to either underperform by generating too many false positives or require large datasets to achieve high accuracy. This affects the effectiveness of DLP tools, as analysts must spend additional time verifying the legitimacy of each detection.
- **Lock of multi-format detection:** Many existing CNN models focus on detecting steganography in specific formats, limiting their adaptability to various types of files, such as audio, video, or different image formats. This results in gaps in coverage, reducing the effectiveness of DLP tools in diverse environments.
- **Integration with DLP tools:** While CNN models for steganalysis are effective in isolation, their integration into existing DLP tools presents a challenge. Most DLP

systems lack built-in steganalysis capabilities, and integration solutions are often not optimized for operational environments, leading to compatibility issues and suboptimal performance.

- **Automated threat response:** While detection models are advancing, there is still a lack of automated workflows in DLP tools for handling steganography-based threats. The absence of such workflows leads to increased manual intervention, slowing down the process of threat mitigation and response.

This layout mirrors the research problem format in your image, focusing on the specific challenges of implementing dual-CNN and feature fusion for steganalysis in DLP tools.

5. Objectives

5.1. Main Objective

The main objective of this research component is to develop an automated and robust steganalysis system within a Data Loss Prevention (DLP) tool to detect steganographic content hidden in images. The system will utilize dual CNN models, operating at both fine and coarse scales, to accurately detect stego images across various formats. Upon detection, the system will block the transfer of these stego images, preventing the leakage of sensitive information. Additionally, the system will automate the threat response actions by integrating customizable workflows for blocking or alerting, providing a clear visual representation of detected threats through a user-friendly GUI. This approach ensures real-time detection, classification, and mitigation of steganographic threats within an enterprise environment.

5.2. Sub Objectives

- **Multi-Scale Feature Extraction**

Develop and implement two CNN models operating at different scales (fine and coarse) to capture both detailed and broad features from images, improving the detection of hidden steganographic content.

- **Real-Time Detection**

Ensure that the dual-CNN steganalysis system operates in real time, enabling the quick identification and blocking of stego images without causing delays in data transfers.

- **Feature Fusion for Improved Accuracy**

Combine the features extracted from both CNN models to create a more comprehensive feature representation, enhancing the overall detection accuracy and reducing false positives.

- **Integration with Data Loss Prevention (DLP)**

Seamlessly integrate the steganalysis detection system into existing DLP frameworks, ensuring compatibility and smooth operation within real-world security infrastructures.

- **Automated Blocking Mechanism**

Design and implement an automated process to immediately block the transfer of detected stego images, preventing unauthorized data leakage.

- **Threat Visualization**

Provide a user-friendly graphical interface that visually represents detection results, allowing security teams to monitor, assess, and respond to steganographic threats efficiently.

- **Support for Multiple Image Formats**

Ensure that the steganalysis system is capable of detecting steganography in a wide range of image formats, increasing its applicability across various use cases and industries.

- **Scalability and Efficiency**

Optimize the system for computational efficiency, enabling it to scale and function effectively in environments with large volumes of data transfers.

5.3. Specific Objectives

- **Building an accurate dual-CNN model** to detect stego images and improve detection accuracy by fusing features from fine and coarse scales.

- **Real-time detection and blocking** of stego images to prevent unauthorized data leakage efficiently without delaying data transfers.
- **Integration and visualization** of the detection results within a user-friendly DLP system, providing clear insights and automated alerts to relevant parties for immediate action.

6. Methodology

6.1. System Architecture

The projects' main goal is to provide an overall solution based on data loss prevention which facilitates in mitigating sensitive data leakage or data leakage attacks in the organization.

The way which the steganalysis component works is illustrated below.

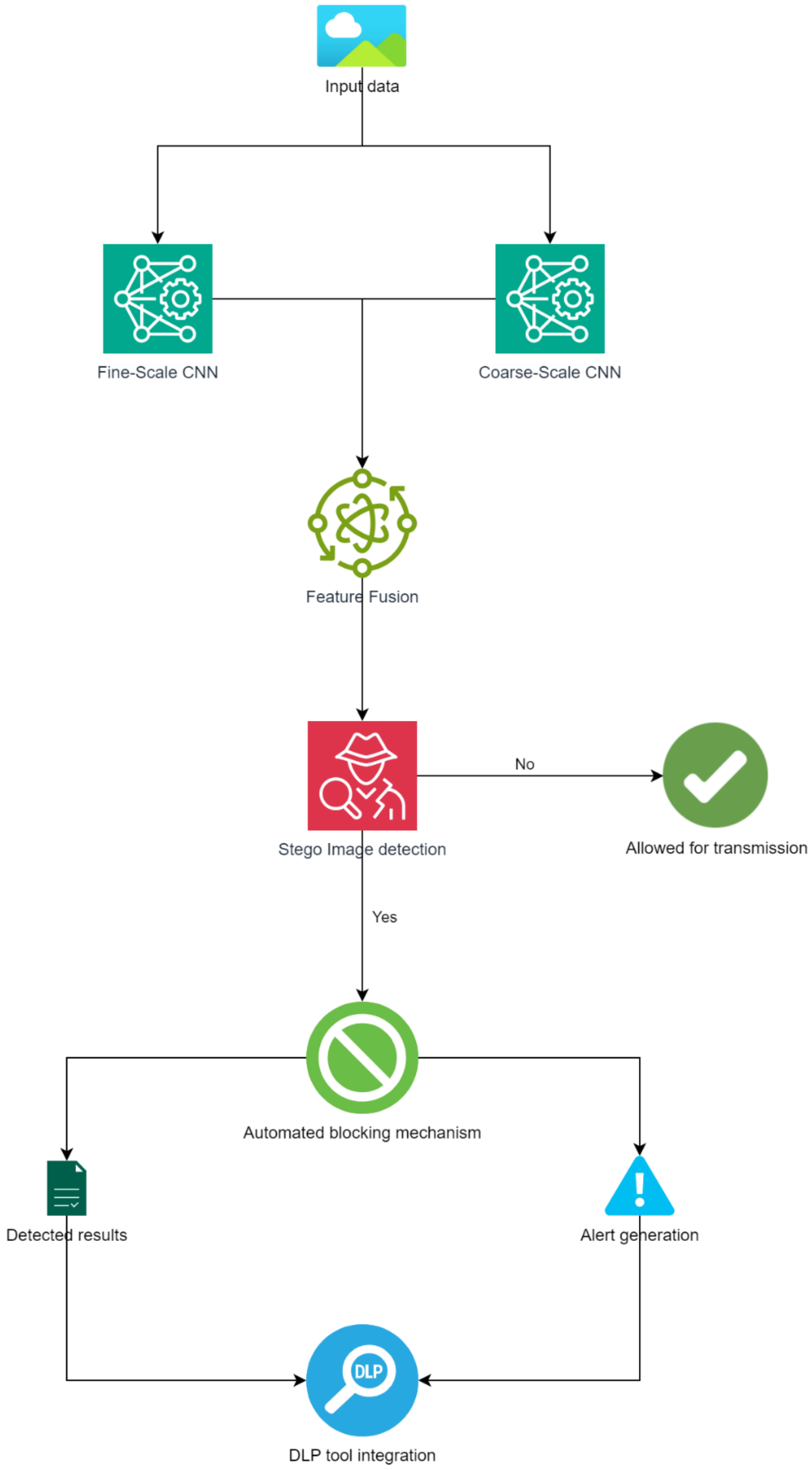

Figure 5 - Individual Component System Architecture

The exact scope of the steganalysis component is the very first issue that needs to be defined, which shall also involve stating the purpose. The main mission here is to find the hidden messages in image files and automatically block their transfer within a DLP system. The understanding of the variety of steganography methodologies at work is vital since the techniques of steganalysis can vary if the data is hidden in an image, audio, video, or any other kind of file. This involves gathering the requirements on types of files the system will be handling, types of steganography detection techniques most applicable, and what the desired results are, such as alert generation and automated blocking.

At the heart of the steganalysis component, this hidden content within an image will be detected using CNNs. One can adopt a dual-CNN architecture where two independently trained CNNs could detect steganographic content at different scales: one CNN for fine-scale features, which captures detailed, low-level patterns present in an image, and another CNN for coarse-scale features that captures high-level, broader patterns. For both, training should be carried out with a dataset containing both clean images and stego images. You can use the publicly available datasets like BOSSbase or ALASKA2 for training purposes.

Each of the CNNs will output features that characterize the content of an image. Such features emanating from both CNNs will be fused together in order to enhance detection accuracy. This step of feature fusion will provide a more robust representation of the image for the model to better detect subtle variations caused by hidden messages.

The trained steganalysis component will be then integrated into the DLP system. Within a DLP environment, all files transferred over the network, such as emails, file uploads or downloads, are constantly monitored in real time. Each time an image file is detected, it will be passed to the steganalysis component for analysis.

The steganalysis component will evaluate the image by the dual-CNN models, and once the system detects steganography, a policy enforcement mechanism will be triggered. This could mean that the DLP tool will block the file transfer or quarantine the file for further review and consideration, or trigger alerts to notify the security team of the possible threat.

For real-time detection, it is necessary to optimize the steganalysis component for speed. This could be done by either reducing the time it takes to process the images or reducing the time for feature extraction. This can be done using model optimization techniques like quantization or pruning, which have minimal effects on model accuracy but come with smaller model sizes and reduced computational complexity. Also, you will need to consider that the DLP system must support high volumes of traffic and be able to pass on image files with efficacy to the steganalysis component.

In order to minimize the need for many manual tasks on the part of the security team, this response automation is very critical. For instance, it should automatically block the transfer once the DLP tool has detected a stego image, or perform similar kinds of actions-such as alerting relevant security personnel. This shall include the integration with automated playbook and customizable workflows.

Apart from blocking file transfers, the system will also be able to issue alerts and logs. The DLP system needs to provide detailed information on each detection event regarding source and destination of the file, nature of steganography detected, and action taken. It will log all these onto a central dashboard, with real-time insights provided to security administrators.

This dashboard should be user-friendly such that an administrator can go into the details of each and every detection, investigate suspicious activities, and adjust system policies if needed. You can also add notifications that send email or SMS notifications in case of critical detections so that immediate action can be taken by the security team.

Before the steganalysis component goes into production, it must be thoroughly tested regarding its system. The testing should also include real scenarios where different steganography techniques will be applied to image files to ensure that many types of hidden messages will be detected by the component. You will also have to test the system for high loads of traffic and evaluate the performance in real-time.

The expected future work will be fine-tuning the system after testing to arrive at the most accurate, fast, and reliable. This most probably is connected with improvements in CNN parameters or even further improvements of feature fusion methods and/or improvements in automation workflows.

6.2. Software solution

Development will be done iteratively; the DLP tool's Agile Software Development Lifecycle using SCRUM methodology will be followed for the development of the steganalysis component. This approach gives much emphasis to continuous integration and collaboration between developers and stakeholders for the efficiency and adaptability of the component.

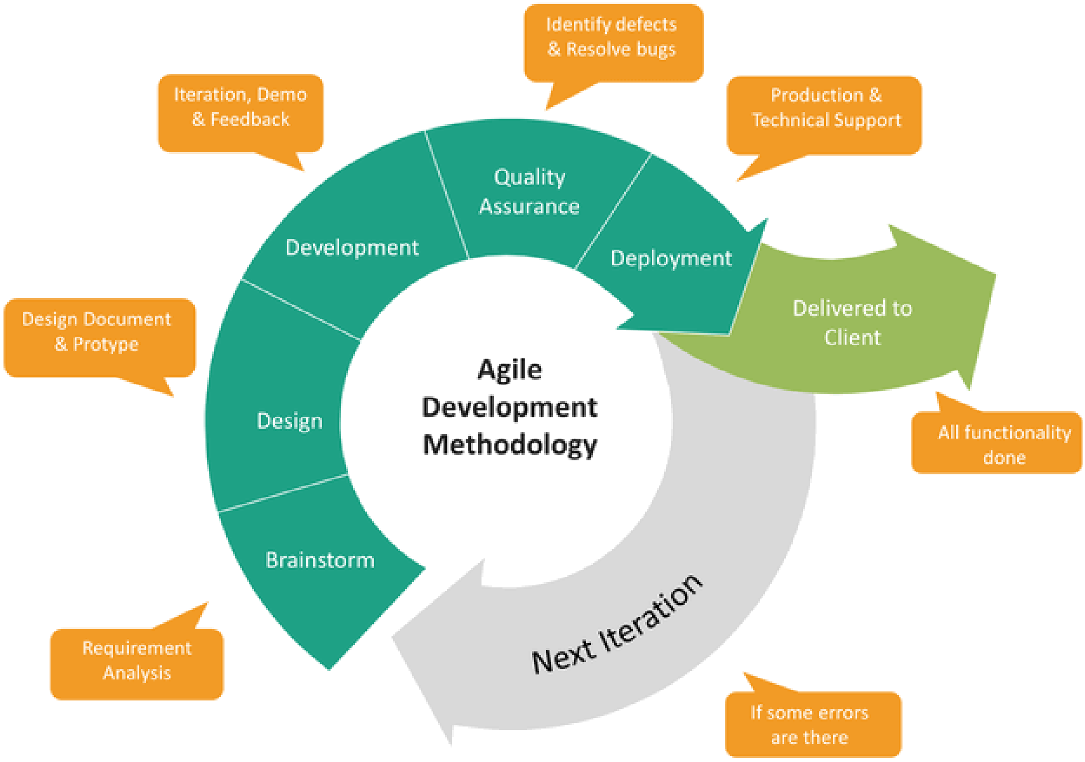

Figure 6-Agile methodology

The solution involves the creation of a dual **Convolutional Neural Network (CNN)** architecture to detect steganographic content embedded in image files. The software will be designed to analyze images for hidden data using two parallel CNN models trained to detect fine-grained and coarse-grained features and automate the detection and blocking of files containing steganographic data in real time, thereby preventing unauthorized data exfiltration.

6.2.1. Requirements gathering and Analysis

This is the most critical stage that will ensure the steganalysis component for the DLP tool will meet all the required objectives. First, the identification of key stakeholders is very much required, including security analysts and developers, followed by gathering inputs on requirements that best suit their needs. The functional requirements include detection of hidden steganographic content in image files, automatic blocking of the detected files, and sending an alert to security teams. The non-functional requirements are real-time performance, accuracy, and scalability to volumes of data.

Datasets need to be collected to train the two CNN models-one for fine-scale and another for coarse-scale detection of steganography. Part of the analysis are also the different techniques of steganography that the system is supposed to detect: spatial and frequency domain methods. This is the phase in which the business and compliance requirements, such as data privacy, are realized to meet the industry standard of the tool. Finally, the gathered requirements will be documented, risks analyzed, and feedback retrieved by cybersecurity experts in order to assure alignment with industry best practices. This structured process lays the ground for an effective and accurate steganalysis system that could be integrated within the DLP tool.

6.2.2. Feasibility study

A technical and economic feasibility study will be conducted to ensure the solution can be implemented efficiently and within the budget. The development team will evaluate the computational resources required to process large volumes of data in real time and the cost-effectiveness of the proposed models.

6.2.3. Dataset Preparation

Two primary datasets will be prepared. A set of images with and without steganographic, labeled appropriately and a dataset of steganographic techniques and hidden patterns to train the dual CNN models.

6.2.4. Implementation

The implementation phase will involve the following steps:

- **Model training** – Develop and train two separate CNN models to detect fine- and coarse-scale features in images. The models will be trained using the labeled dataset.
- **Feature Fusion** – Combine the outputs from both CNNs to generate a unified representation, which will enhance the detection accuracy.
- **Real-Time Detection** – Integrate the model into the DLP system to analyze images in real time and block transfers of files that contain hidden data.

6.2.5. Testing

To ensure the software functions as intended, the following tests will be conducted:

- Unit testing – Individual components of the steganalysis tool will be tested for errors.
- Integration testing – Ensure that the steganalysis component integrates smoothly into the larger DLP solution.
- System testing – The entire DLP system, with the integrated steganalysis component, will undergo comprehensive testing.
- User acceptance testing (UAT) – End users, such as security analysts, will test the system to ensure it meets their operational needs.

6.2.6. Deployment

The system will be deployed in a real-time environment using a cloud-based infrastructure. The dual CNN models will be hosted on a server that supports the processing power required for image analysis at scale.

6.3. Tools and Techniques

Tools

- **Python**: As the primary programming language for implementing machine learning models and system integration.
- **Anaconda**: For managing the machine learning environment and dependencies, especially during model development.
- **PyCharm**: For Python development, providing a robust Integrated Development Environment (IDE).
- **Flask/Django**: Tentatively for developing the web-based interface to visualize steganalysis detection results.
- **Heroku/AWS/GCP**: For cloud deployment, allowing for scalability and easy integration with a DLP system.
- **Jupyter Notebooks**: For training and experimenting with different machine learning models.
- **GitHub**: For version control and collaboration during the development process.
- **Microsoft Planner**: For project management and task tracking.

Techniques

- **Convolutional Neural Networks (CNNs)**: To train two separate models for detecting steganographic content at different image scales.
- **Feature Fusion**: To combine fine and coarse-scale features from the CNN models, improving detection accuracy.
- **Natural Language Processing (NLP)**: Optional, if text-based steganalysis or threat feed parsing is required.
- **Data Augmentation**: To enhance the dataset with synthetic images for better model training.
- **Testing and Validation**: Includes unit testing, integration testing, and user acceptance testing (UAT) to ensure system quality and functionality.

7. Project Requirements

7.1. Functional Requirements

- The system shall detect hidden steganographic content within image files in real time using dual Convolutional Neural Networks (CNNs) operating at fine and coarse scales.
- Upon detection of steganographic content, the system shall automatically block the transfer of the identified image to prevent unauthorized data leakage.
- The system shall implement a feature fusion process that combines outputs from both CNN models to enhance detection accuracy and reduce false positives.
- The system shall analyze various image file formats, including but not limited to JPEG, PNG, and BMP, to ensure comprehensive coverage.
- The system shall generate real-time alerts when a stego image is detected, providing details such as file name, source, destination, and confidence level.
- Notifications shall be sent to designated security personnel via email, SMS, or integrated messaging systems.
- All detection events, actions taken, and system activities shall be logged for auditing purposes.

- The system shall generate periodic reports summarizing detection statistics, trends, and system performance.
- A user-friendly dashboard shall display real-time detection events, system status, and allow for interactive data visualization.
- Security analysts shall be able to view, filter, and analyze detection data through the interface.

7.2. Non-Functional Requirements

- The system should handle high volumes of file transfers and data streams without performance degradation, ensuring scalability for large organizations.
- The detection system should maintain an accuracy rate above 95% for identifying steganographic content while minimizing false positives and negatives.
- The system must be able to scale horizontally to accommodate increasing data loads, such as higher volumes of image transfers across different environments.
- The system must be reliable in detecting steganographic content and must function continuously without frequent crashes or downtime.
- The system must ensure data privacy and security, with encryption used for data both in transit and at rest.
- Access control must be enforced using role-based access to ensure that only authorized users can manage and view detection results and system configurations.
- The system must have a user-friendly interface that allows security analysts and administrators to easily interact with the system, review detection logs, and configure system settings.
- The system must be easy to maintain, with modular components that can be updated, patched, or replaced without causing significant downtime.
- It should support regular updates to the steganalysis models and DLP rules to adapt to evolving threats and new steganography techniques.

7.3. System Requirements

The system requirements outline the necessary software, hardware, and resources needed to ensure the proper functioning of the steganalysis component within the DLP solution. Below are the key requirements for the project:

- **Python:** The primary programming language used for implementing machine learning models and system logic.
- **Anaconda**: To manage virtual environments and handle dependencies for the machine learning frameworks.
- **PyCharm or VS Code**: For code development and testing in a well-supported Integrated Development Environment (IDE).
- **Flask or Django**: (Optional) Tentative frameworks for developing the web-based user interface for visualization and reporting.

7.4. User Requirements

Security analysts need the system to provide real-time detection of steganographic content in image files and alert them immediately upon detection. The system should feature a user-friendly dashboard that offers a comprehensive view of detection events, system health, and recent alerts. Analysts also need detailed reports that provide insights into detected stego content, file metadata, and actions taken by the system. Additionally, they require the ability to configure alerts (e.g., email or SMS notifications) and set thresholds for detection confidence levels to prioritize critical alerts.

For **incident response teams**, the system must automatically block suspicious files to prevent data exfiltration, eliminating the need for manual intervention. Access to detailed logs is crucial, as it allows teams to analyze incidents, understand the context of the detected threats, and evaluate system performance. The tool must integrate with existing workflows, providing customizable options for automating incident response actions, such as quarantining files or notifying specific team members.

System administrators require control over user access, allowing them to manage permissions and roles within the system. They also need the ability to configure system settings, including detection parameters and system thresholds, to optimize performance. Administrators must have an intuitive interface to manage system updates, patches, and perform troubleshooting to ensure smooth operation.

For **compliance officers**, the system should generate audit logs that track all detection events and responses in a tamper-proof manner, ensuring that the organization meets data protection and regulatory standards. Additionally, the system must offer exportable reports that can be used for compliance audits, providing a clear record of detected incidents, responses, and overall system effectiveness.

8. References

[1] Liu, S., & Kuhn, R. (2010). Data loss Prevention. *IT Professional*, *12*(2), 10–13. https://doi.org/10.1109/mitp.2010.52

[2] Sloan, T., & Hernandez-Castro, J. (2015). Forensic analysis of video steganography tools. *PeerJ Computer Science*, *1*, e7. https://doi.org/10.7717/peerj-cs.7

[3] Atawneh, S., Almomani, A., & Sumari, P. (2013). Steganography in digital images: Common approaches and tools. *IETE Technical Review*, *30*(4), 344–358. https://doi.org/10.4103/0256-4602.116724

[4] Gutiérrez-Cárdenas, J. M. (2017). Steganography and Data Loss Prevention: An overlooked risk? *International Journal of Security and Its Applications*, *11*(4), 71–84. https://doi.org/10.14257/ijsia.2017.11.4.06

[5] Wikipedia contributors. (2024, August 17). *Steganography*. Wikipedia. https://en.wikipedia.org/wiki/Steganography

[6] Chaumont, M. (2019). Deep Learning in steganography and steganalysis from 2015 to 2018. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.1904.01444

[7] Dwaik, A., Belkhouche, Y. (2022). Analysis of Deep Learning-Based Image Steganalysis Methods Under Different Steganographic Algorithms. In: Bebis, G., *et al*. Advances in Visual Computing. ISVC 2022. Lecture Notes in Computer Science, vol 13599. Springer, Cham. https://doi.org/10.1007/978-3-031-20716-7_22

[8] Fayayola, N. O. A., Olorunfemi, N. O. L., & Shoetan, N. P. O. (2024). DATA PRIVACY AND SECURITY IN IT: A REVIEW OF TECHNIQUES AND CHALLENGES. *Computer Science & IT Research Journal*, *5*(3), 606–615. https://doi.org/10.51594/csitrj.v5i3.909

[9] Cooper, S., & Cooper, S. (2024, March 21). *The best data loss Prevention software Tools*. Comparitech. https://www.comparitech.com/data-privacy-management/data-loss-prevention-tools-software/

[10] Dalal, M., Juneja, M. Steganography and Steganalysis (in digital forensics): a Cybersecurity guide. *Multimed Tools Appl* **80**, 5723–5771 (2021).

[11] Kheddar, H., Hemis, M., Himeur, Y., Megías, D., & Amira, A. (2024). Deep learning for steganalysis of diverse data types: A review of methods, taxonomy, challenges and future directions. *Neurocomputing*, *581*, 127528. https://doi.org/10.1016/j.neucom.2024.127528

[12] Progonov, D. (2023). Destruction of stego images formed by adaptive embedding methods with dictionary learning methods. *Theoretical and Applied Cybersecurity*, *4*(1). https://doi.org/10.20535/tacs.2664-29132022.1.254883

[13] Bhuva, B. D., Zavarsky, P., & Butakov, S. (2021). An analysis of effectiveness of StegoAppDB and data hiding efficiency of StegHide Image Steganography tools. *2021 2nd International Conference on Secure Cyber Computing and Communications (ICSCCC)*. https://doi.org/10.1109/icsccc51823.2021.9478123

[14] De La Croix, N. J., & Ahmad, T. (2023). Toward secret data location via fuzzy logic and convolutional neural network. *Egyptian Informatics Journal*, *24*(3), 100385. https://doi.org/10.1016/j.eij.2023.05.010

[15] Monika, A., & Eswari, R. (2022). Prevention of hidden information security attacks by neutralizing Stego-Malware. *Computers & Electrical Engineering*, *101*, 107990. https://doi.org/10.1016/j.compeleceng.2022.107990

[16] Hidayasari, N., Riadi, I., & Prayudi, Y. (2020). Steganalysis Using Yedrodj-net net's Convolutional Neural Networks (CNN) Method on Steganography Tools. *Proceeding International Conference on Science and Engineering*, *3*, 207–211. https://doi.org/10.14421/icse.v3.499

[17] Kuznetsov, A., Luhanko, N., Frontoni, E. *et al.* Image steganalysis using deep learning models. *Multimed Tools Appl* **83**, 48607–48630 (2024). https://doi.org/10.1007/s11042-023-17591-0

[18] Nicolás-Sánchez, A., Castro-Toledo, F.J. Uncovering the social impact of digital steganalysis tools applied to cybercrime investigations: a European Union perspective. *Crime Sci* **13**, 11 (2024).

[19] Farooq, N., Selwal, A. Image steganalysis using deep learning: a systematic review and open research challenges. *J Ambient Intell Human Comput* **14**, 7761–7793 (2023).

[20] Fajembola, J. N. U. T. T. O. S. E. a. a. K. V. (2024, April 30). *Steganalysis of an Image-Based Semantic Segmented and Binary Pattern Complex STEGO file using RNN and DenseNet*. https://www.fjpas.fuoye.edu.ng/index.php/fjpas/article/view/308