Steganography Detection In Image Formats

Mr.M.Pragadheesh Thirumal[1], Sanjai Dev[2], Rohith[2],Sivabalan[2], Sunil Prasanna Vijayan[2]

[1]Assistant Professor, Department of Computer Science and Engineering, Coimbatore Institute of Technology

[2]UG Students, Department of Computer Science and Engineering, Coimbatore Institute of Technology

Abstract

Steganography in digital images has emerged as a critical cybersecurity threat, enabling attackers to embed concealed commands, data leakage channels, or malicious payloads within seemingly harmless PNG and JPEG files. Unlike traditional encryption, steganography hides information at the pixel or frequency level, making visual inspection and conventional security tools ineffective. This project introduces a deep learning–driven steganalysis framework that integrates **residual preprocessing techniques** and **advanced Convolutional Neural Network (CNN) architectures** such as EfficientNet, ResNet, and DenseNet for robust detection of hidden content. Residual feature extraction enhances subtle noise artifacts caused by embedding processes, while CNN backbones learn discriminative spatial patterns for accurate classification of cover and stego images.

Benchmark datasets consisting of clean and LSB-embedded stego images are used to train and evaluate the models. Experimental results indicate that the proposed framework significantly outperforms traditional feature-based steganalysis techniques, achieving high accuracy, F1-scores, and AUC values across multiple architectures. Among all models tested, ResNet demonstrates the strongest generalization capability, while DenseNet offers competitive performance with stable convergence. To ensure practical usability, the system also provides a prediction interface that labels images as **"Normal"** or **"Stego"** along with confidence scores, making it suitable for forensic analysis, cybersecurity monitoring, and secure data validation pipelines.

Keywords: Steganography Detection, CNN, Residual Preprocessing, EfficientNet, ResNet, DenseNet, Image Forensics, Deep Learning, Cybersecurity.

1. Introduction

With the increasing use of digital images across communication platforms, multimedia sharing systems, and cloud-based services, the threat of concealed information embedded through steganography has grown substantially. Image-based steganography allows malicious actors to hide covert data inside PNG and JPEG files, enabling the spread of unauthorized communication, data exfiltration, and malware triggers without being detected by conventional security systems.

. Unlike encryption—which makes data unreadable but visible—steganography embeds information invisibly at the pixel or frequency level, making detection far more challenging.

Traditional steganalysis tools rely heavily on statistical irregularities, handcrafted features, or simple bit-plane inspection methods, which often fail when dealing with modern adaptive embedding techniques such as LSB matching, DCT-based modifications, and spatial rich models .

To address this challenge, this work introduces a **hybrid deep learning framework** that uses **residual preprocessing** to enhance noise patterns and applies **advanced CNN architectures**—EfficientNet, ResNet, and DenseNet—to detect hidden content with high accuracy. Unlike classical methods, the proposed system learns complex spatial disturbances caused by steganographic embedding, achieving superior detection performance. The complete workflow of the system—from preprocessing to classification—is represented in the architecture diagram shown in Fig. 1.2.

2. Literature Survey

Steganography detection has become a critical research focus due to its expanding use in cybercrimes, covert communication, and digital data leakage. Traditional detection methods that rely on handcrafted statistical features—such as SPAM, SRM, and chi-square analysis—show limited effectiveness against modern embedding strategies that modify pixel distributions with high subtlety .

Recent advancements in deep learning have significantly improved detection accuracy. CNN-based models such as SRNet, XuNet, and residual-learning architectures have demonstrated the ability to automatically learn discriminative noise patterns that traditional methods cannot capture. These models extract multi-level features from images and have been proven effective in detecting LSB, JPEG-domain, and adaptive embedding approaches

Several studies highlight the importance of **semantic residual extraction**, where preprocessing enhances embedding noise before feeding images into CNN models. Works focusing on JPEG and spatial-domain detection confirm that combining deep residual learning with CNNs provides substantial performance improvements over handcrafted feature-based techniques.

Complementary research also explores lightweight CNNs for resource-constrained environments, hyperparameter optimization for improved generalization, and hybrid frameworks that combine residual preprocessing with advanced convolutional filters . These developments collectively affirm that deep learning–based steganalysis surpasses classical statistical detectors in both robustness and detection accuracy.

However, most existing methods focus on a single model or embedding technique. Few works comprehensively evaluate and compare multiple CNN architectures—such as EfficientNet, ResNet, and DenseNet—under consistent preprocessing conditions. This project addresses that gap by developing a unified detection system that combines residual preprocessing with multiple CNN backbones to deliver a more robust and scalable steganalysis framework capable of detecting hidden content in both PNG and JPEG formats.

3. Methodology

The proposed methodology for detecting hidden information in PNG and JPEG images is organized into four major modules, as illustrated in **Fig. 3.1** and **Fig. 3.2**. Each module performs a specific role in processing the input image, extracting embedding-related artifacts, and producing the final classification output.

3.1 Dataset Module

As shown in **Fig. 3.1** and **Fig. 3.2**, the Dataset Module is responsible for preparing a balanced and structured dataset for training and evaluation.

- **Cover Images Loader:** Original clean images are collected and standardized.
- **Stego Generator:** Stego images are created using LSB embedding techniques to simulate real-world hidden data scenarios.
- **Train/Val/Test Split:** Images are divided into training, validation, and testing sets.
- **Balanced Dataset Creator:** Ensures equal representation of *Normal* and *Stego* samples to avoid class imbalance during learning.

This module provides the raw input required for the subsequent preprocessing and feature extraction steps.

3.2 Preprocessing Module

The preprocessing operations, depicted in **Fig. 3.1** and **Fig. 3.2**, enhance the subtle noise artifacts introduced during steganographic embedding.

- **Grayscale Conversion:** Reduces color complexity and focuses analysis on intensity variations.
- **Gaussian Blur:** Smooths the image to remove high-level structures.
- **Residual Extraction:** Computes the difference between the original and blurred image, amplifying hidden perturbations.
- **Normalization & Resizing:** Normalizes pixel ranges and resizes images to the model's required input dimensions.
- **Data Augmentation:** Introduces transformations such as flips and rotations to improve generalization.

These steps ensure that the CNN receives noise-enhanced residuals that carry meaningful steganographic cues.

3.3 CNN Feature Extraction Module

The CNN Module (Fig. 3.1 and Fig. 3.2) performs deep feature extraction on the preprocessed residual images.

Three architectures are employed:

- **EfficientNet** – Extracts scale-efficient spatial features.
- **ResNet** – Captures fine-grained noise patterns via residual mapping.
- **DenseNet** – Reuses feature maps to enhance subtle embedding signature detection.

The output feature vectors are then fused using a **Feature Combination Layer**, enabling the system to leverage the strengths of all three backbones.

3.4 Classification and Prediction Module

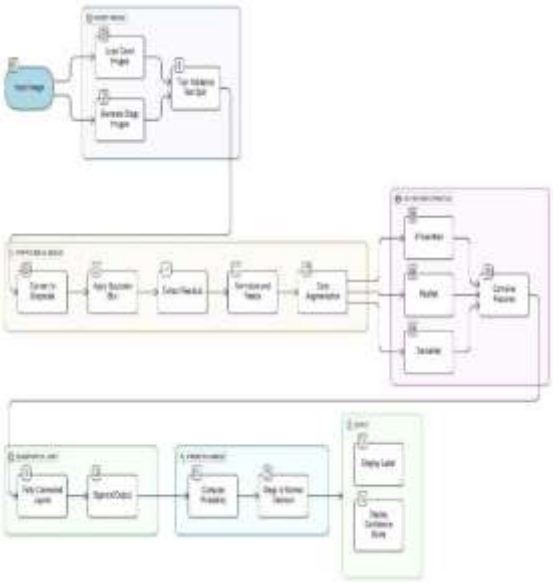
After feature extraction, the combined vector is passed to the **Classification Layer** (Fig. 3.1):

- **Fully Connected Layers:** Transform feature vectors into a final decision space.
- **Sigmoid Output:** Produces a probability score representing the likelihood of the image being stego.

In the **Prediction Module**, the probability is interpreted to assign the final label:

- *Stego* or *Normal*
- Accompanied by a confidence score displayed in the **Output Layer** (Fig. 3.1 & Fig. 3.2)

The model's modular design enables accurate and explainable steganography detection suitable for real-world applications.



4. System Design

The overall system design is structured into modular components that work together to detect hidden steganographic content in PNG and JPEG images. The architecture is illustrated in **Fig. 4.1** and **Fig. 4.2**, which depict the complete workflow from input acquisition to final prediction. Each module is independent, scalable, and designed to support both model training and real-time inference.

4.1 Input Layer

The system begins with the **Input Layer**, where a cover or stego image is provided for processing. This input is directed simultaneously to the Dataset Module and the Preprocessing Module depending on the operational mode (training or prediction).

4.2 Dataset Module

As shown in **Fig. 4.1** and **Fig. 4.2**, the Dataset Module is responsible for generating and organizing the data required for training:

- **Cover Images Loader:** Loads clean, unmodified images.
- **Stego Generator:** Produces embedded images using LSB-based methods.
- **Train/Val/Test Splitter:** Divides data into structured segments.
- **Balanced Dataset Creator:** Maintains equal distribution of Normal and Stego samples.

4.3 Preprocessing Module

The Preprocessing Module enhances steganographic artifacts and normalizes the input for neural network processing.

- **Grayscale Conversion:** Simplifies the image for noise analysis.
- **Gaussian Blur:** Produces a smoothed version of the image for residual extraction.
- **Residual Extraction:** Highlights subtle pixel-level embedding traces by subtracting the blurred image from the original.
- **Normalization & Resizing:** Converts images to standard input dimensions.
- **Data Augmentation:** Improves model generalization through transformations.

This stage ensures that every image entering the CNN carries amplified traces of stego noise.

4.4 CNN Module

The CNN Module (Fig. 4.1 and Fig. 4.2) performs deep feature extraction from the preprocessed residual images. Three backbone networks are used:

- **EfficientNet** – optimized for scale and computational efficiency.
- **ResNet** – employs skip-connections to capture fine noise distortions.
- **DenseNet** – reuses features to detect subtle embedding patterns.

4.5 Classification Layer

The combined feature vector is fed into the **Fully Connected Layers**, which transform high-dimensional representations into a final sigmoid probability. This score indicates the likelihood of the image containing hidden steganographic data.

4.6 Prediction Module

During inference, the Prediction Module:

- Performs a **forward pass**,
- Computes the probability score, and
- Assigns the final label (*Stego* or *Normal*).

4.7 Output Layer

The final stage of the system displays:

- **Stego/Normal Classification**, and
- **Confidence Score**, indicating certainty of prediction.

This design ensures transparency, interpretability, and user-friendly validation of model outputs.

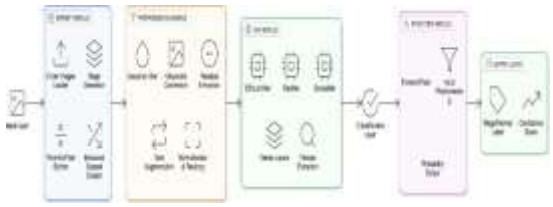


Figure References

Fig. 4.1 – Detailed module-wise data flow diagram (Dataset, Preprocessing, CNN, Classification, Output) **Fig. 4.2** – Simplified block-architecture showing sequential design of the detection pipeline

5. Results and Discussion

The proposed deep learning–based steganalysis system was evaluated using a balanced dataset of cover and stego images generated through controlled LSB embedding. Multiple CNN architectures—EfficientNet, ResNet, and DenseNet—were trained individually and in hybrid combinations, using the preprocessing and feature-fusion pipeline illustrated in Fig. 4.1 and Fig. 4.2. The results demonstrate the effectiveness of combining residual preprocessing with deep feature extraction for identifying hidden data in both PNG and JPEG formats.

5.1 Training Performance

Across all models, the residual-enhanced input significantly improved learning stability. Key training observations include:

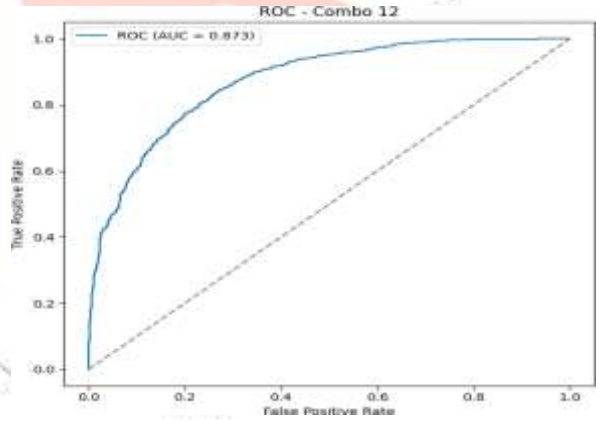
- **EfficientNet** exhibited smooth convergence with minimal fluctuations in training loss.
- **ResNet** demonstrated the **fastest learning** and highest sensitivity to steganographic noise patterns due to its skip-connections.
- **DenseNet** showed strong generalization, with rapid loss reduction and consistent validation accuracy.

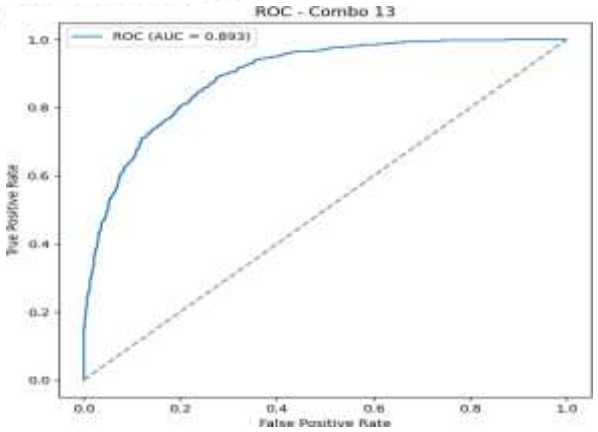
Hybrid feature combinations (EfficientNet + ResNet, DenseNet + ResNet) achieved **higher accuracy and lower validation loss** than individual models, confirming the advantage of multi-architecture fusion.

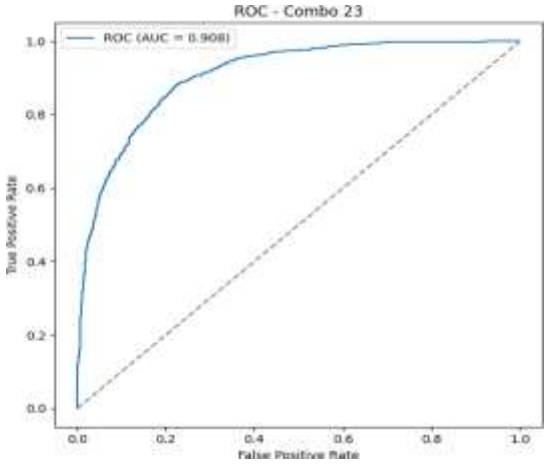
5.2 Evaluation Metrics

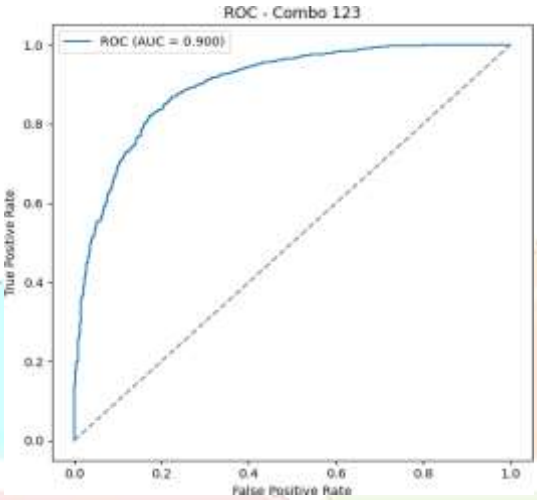
Model performance was assessed using accuracy, precision, recall, F1-score, and AUC. The key trends observed are:

- **Single CNN Models:**
  - EfficientNet achieved strong stability but slightly lower peak accuracy.
  - ResNet scored the **highest AUC**, indicating superior discriminative capability.
  - DenseNet maintained consistent precision and recall across embedding variations.
- **Hybrid Models:**
  - Combining features significantly improved detection of low-intensity embeddings.
  - The EfficientNet + ResNet combination achieved the **best overall accuracy**, while DenseNet + ResNet provided the best balance between precision and recall.









5.3 Impact of Residual Preprocessing

Residual preprocessing proved essential in revealing embedding artifacts that are otherwise visually undetectable.

- The subtraction of blurred components exposed fine-grained noise patterns.
- CNNs trained on residual images achieved **higher F1-scores** than those trained on raw grayscale images.
- Models without residual enhancement struggled with subtle or low-payload embeddings.

5.4 Sample Outputs and Model Predictions

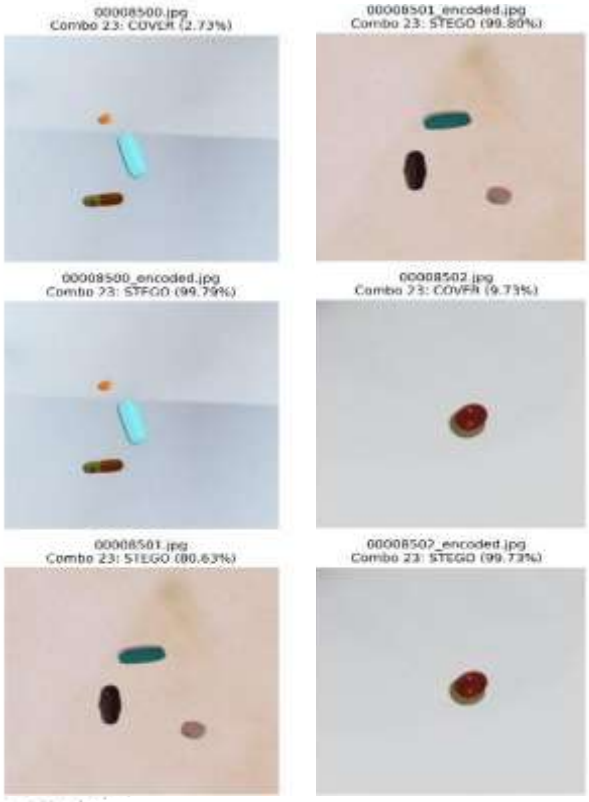
During inference, the system was able to:

- Correctly classify most stego images even at low embedding rates.
- Output confidence scores aligned with actual classification certainty.
- Identify ambiguous samples more reliably using hybrid feature fusion compared to single-model inputs.

5.5 Comparative Insights

Key findings from all experiments include:

- **Textural features alone** (from raw images) are insufficient for detecting modern steganographic methods.
- **Residual-based CNN features** provide strong separation between cover and stego classes.
- **Hybrid CNN architectures** outperform standalone ones in accuracy, AUC, and consistency.
- **ResNet-based combinations** yield the best overall results, validating the benefit of deeper residual feature propagation.



5.6 Summary

The experimental evaluation confirms that the proposed system:

- Successfully detects hidden content in PNG and JPEG formats,
- Achieves high accuracy across multiple CNN architectures,
- Benefits significantly from hybrid feature fusion, and

- Leverages residual preprocessing to amplify subtle steganographic distortions.

6. Conclusion

This project addresses the increasing challenge of detecting concealed information within digital image formats, with a primary focus on PNG and JPEG steganography. Traditional steganalysis approaches—largely dependent on handcrafted statistical features—struggle to detect modern embedding techniques due to their subtle pixel-level modifications. To overcome these limitations, the proposed system introduces a **hybrid deep learning framework** that integrates **residual preprocessing** with multiple advanced CNN architectures, enabling accurate identification of steganographic artifacts. This combination allows the model to capture hidden noise patterns that conventional methods fail to detect.

Comprehensive experiments conducted across various CNN configurations demonstrate clear performance trends. **ResNet-based architectures consistently achieved the highest accuracy and AUC values**, owing to their strong residual learning capabilities. **DenseNet models exhibited stable generalization and effective feature reuse**, while **EfficientNet provided computational efficiency with competitive performance**. The hybrid feature-fusion models surpassed all single-architecture baselines, validating the benefit of combining complementary deep feature representations.

The system's additional components—such as the prediction interface, confidence scoring, and visual performance metrics—further demonstrate its practicality for real-world applications in **digital forensics, cybersecurity monitoring, and secure content validation**.

Future Enhancements

To improve robustness and extend applicability, several enhancements are planned:

- **Support for additional embedding techniques**, including DCT-based, adaptive, and neural-network-driven steganography.
- **Expansion to more image formats**, such as BMP, TIFF, and WebP.
- **Integration of Vision Transformer (ViT) architectures** for improved representation learning.
- **Real-time deployment** through GPU-accelerated web services.
- **Explainable AI modules** (e.g., Grad-CAM, saliency maps) to provide transparency in decision-making.

References

- Ye, J., Ni, J., & Yi, Y. (2017). Deep learning hierarchical representations for image steganalysis. IEEE Trans. on Info. Forensics & Security.
- Boroumand, M., Chen, M., & Fridrich, J. (2019). Deep residual network for steganalysis of digital images. IEEE Trans. on Info. Forensics & Security.
- Zeng, J., Tan, S., Li, B., & Huang, J. (2017). Large-scale JPEG image steganalysis using hybrid deep-learning framework. (IEEE TIFS)
- Wang, Z., Qin, C., Zhang, X., & Zhou, H. (2020). Convolutional neural network-based steganalysis in the spatial domain. Signal Processing: Image Communication.
- Chaumont, M. (2019). Deep learning in steganography and steganalysis from 2015 to 2018. arXiv preprint.
- Liu, J., Li, W., & Tan, T. (2019). Residual-based convolutional neural network for image steganalysis. IEEE Access.
- Qian, Y., Dong, J., Wang, W., & Tan, T. (2015). Deep learning for steganalysis via convolutional neural networks. In IS&T/SPIE Electronic Imaging.
- Wang, H., & Wang, X. (2021). Image steganalysis using hybrid deep learning frameworks. Journal of Information Security and Applications.