E-NetSRM: Exploring Preprocessing Strategies for Steganalysis

Adrian Mauricio-Gonzalez
Department of Computer Science
University of Central Florida
ad375233@ucf.edu

Abstract

Steganography, the practice of embedding hidden messages within digital media, poses cybersecurity risks by enabling covert communication and malware transmission. Steganalysis aims to detect these hidden messages but struggles with subtle steganographic modifications. This study introduces E-NetSRM, an enhanced EfficientNet-based steganalysis model incorporating preprocessing and architectural refinements. This study integrates a high-pass filtering layer using Spatial Rich Model (SRM) kernels to emphasize noise residuals, introduces a Siamese modification for pairwise feature comparison, and modifies the EfficientNet input stem and backbone for improved low-level feature extraction. Evaluated on the ALASKA2 dataset, E-NetSRM outperforms baseline models, including CCNet, SiaStegNet, and EfficientNet variants. An ablation study highlights the benefits of high-pass filtering and architectural modifications, though the Siamese approach presents training instability. Key contributions include the development of E-NetSRM, an in-depth ablation analysis, and the open-source release of CCNet. This study emphasizes the importance of preprocessing and model refinements in improving deep learning-based steganalysis. The code is available at https://github.com/amauriciogonzalez/e-netsrm

1. Introduction

Steganography is the practice of concealing information within digital media, often in a manner that renders the presence of hidden data imperceptible to the human eye. This technique is commonly employed for secure communication, watermarking, and digital rights management. However, it also raises significant concerns in cybersecurity, as it can be exploited to secretly transmit malicious content while bypassing traditional detection methods. Steganalysis, the countermeasure to steganography, involves detecting and classifying stego images—images that contain hidden messages—against cover images, which are unaltered. The challenge in steganalysis lies in identifying subtle modifications in pixel distributions, particularly when sophisticated embedding techniques minimize detectable artifacts.

There are several reasons why this topic was chosen for research. Steganography remains an evolving threat in cybersecurity, demanding continuous advancements in detection techniques. Additionally, steganalysis is a unique problem in computer vision, as it relies on identifying imperceptible lower-level features rather than high-level semantic features typically used in scene understanding. Despite the importance of steganalysis, many existing lightweight models, such as EfficientNet [1], are primarily used by simply fine-tuning pretrained architectures without extensive exploration of preprocessing techniques or architectural modifications. This work seeks to address these gaps by investigating methods to enhance EfficientNet's performance in steganalysis, specifically through preprocessing enhancements and structural refinements.

In this study, multiple models are evaluated, including CCNet [2], SiaStegNet [3], and variations of EfficientNet, using the ALASKA2 dataset [4], a large-scale steganalysis dataset containing cover images and images embedded with hidden data using different steganographic algorithms. A subset of this dataset was used, which consists of 60,000 training samples and 1,000 evaluation samples, evenly distributed across four classes: cover, JMiPOD [5], JUNIWARD [6], and UERD [7]. The results indicate that EfficientNet, when enhanced with a high-pass filter and an architectural modification, achieves the best overall performance. This variant, referred to as E-NetSRM, attains a binary accuracy of 68.90% and an F1 score of 0.5627, outperforming baseline models. The ablation study further confirms that preprocessing and structural refinements significantly enhance detection performance, whereas the Siamese modification, while promising, suffers from unstable training dynamics.

The key contributions of this work are as follows:

- The proposal of E-NetSRM, an enhanced EfficientNet-based model incorporating high-pass filtering and architectural modifications, achieving exceptional performance on the ALASKA2 dataset.

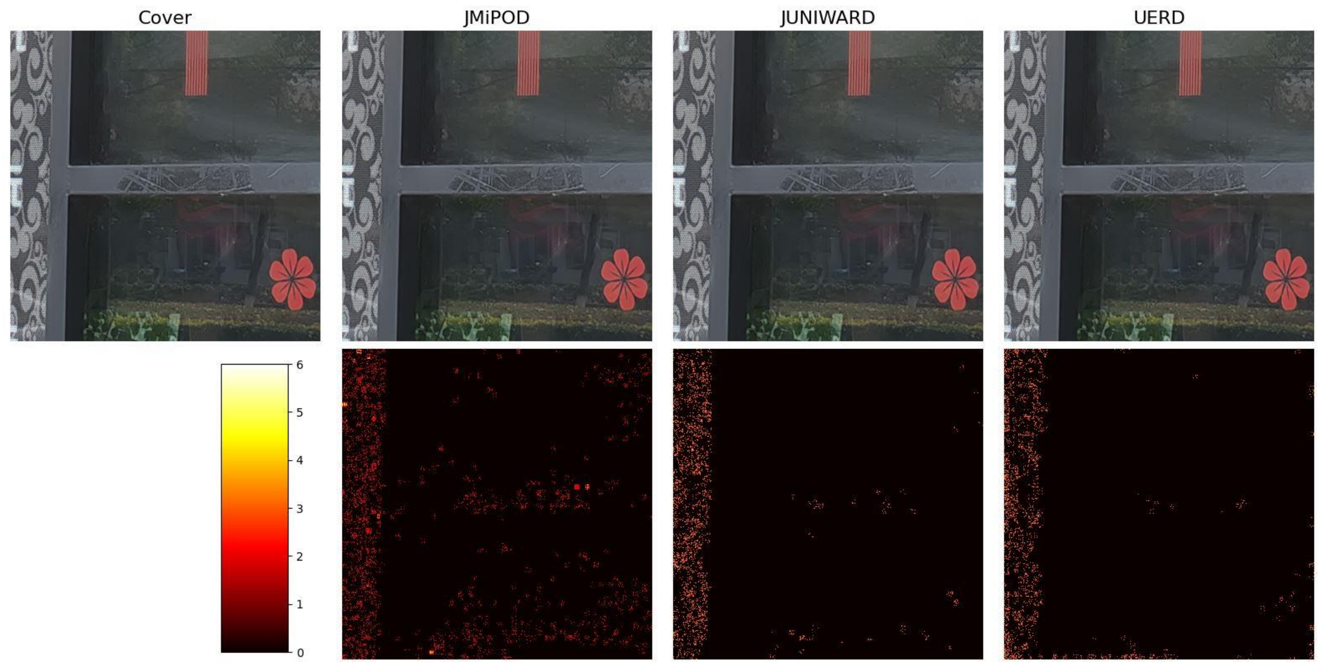

Figure 1: A cover image along with the application of the JMiPOD, JUNIWARD, and UERD steganographic algorithms (first row) with heatmaps visualizing the pixel differences between the associated stego image and cover image in grayscale (second row).

- A comprehensive ablation study to analyze the impact of preprocessing and structural modifications on steganalysis performance.
- An open-source implementation of CCNet.

2. Related Work

This section provides an overview of steganographic algorithms used in the ALASKA2 dataset and existing deep learning-based steganalysis models.

2.1. Steganographic Algorithms

Steganographic techniques embed hidden information within digital images while minimizing perceptible artifacts. The ALASKA2 dataset incorporates three widely used spatial-domain steganographic algorithms: JMiPOD, JUNIWARD, and UERD.

JMiPOD (J-UNIWARD with Minimizing the Probability of Detection) is a steganographic algorithm that optimizes embedding by minimizing a detectability function. It strategically distributes message bits to reduce statistical artifacts that steganalysis models might detect.

JUNIWARD (J-UNIversal WAvelet Relative Distortion) operates in the wavelet domain and leverages a distortion function to determine optimal embedding locations. By modifying coefficients in textured and edge regions, it enhances security against steganalysis while maintaining image quality.

UERD (Uniform Embedding Revisited Distortion) applies a uniform embedding strategy that minimizes changes to pixel values while distributing modifications across the image. This approach reduces detectability while maintaining a balance between robustness and imperceptibility.

These steganographic techniques aim to embed information while preserving image integrity, making their detection a challenging problem that requires advanced deep learning-based steganalysis methods. Figure 1 illustrates how each algorithm alters a sample from the ALASKA2 dataset. The corresponding heatmaps visualize pixel differences between the stego and cover images in grayscale, highlighting that the modifications are subtle—primarily ranging from 0 to 6 in intensity in this case. This minimal alteration underscores the imperceptibility of the embedding process, reinforcing the difficulty of detecting steganographic content through simple visual inspection and the importance of studying the effects of different preprocessing techniques.

2.2. Steganalysis Models

Deep learning-based steganalysis models have demonstrated significant improvements over traditional handcrafted feature-based methods. In this study, three existing models are evaluated—CCNet, SiaStegNet, and EfficientNet—as baselines for comparison.

CCNet is a convolutional neural network designed specifically for spatial-domain steganalysis. It employs a

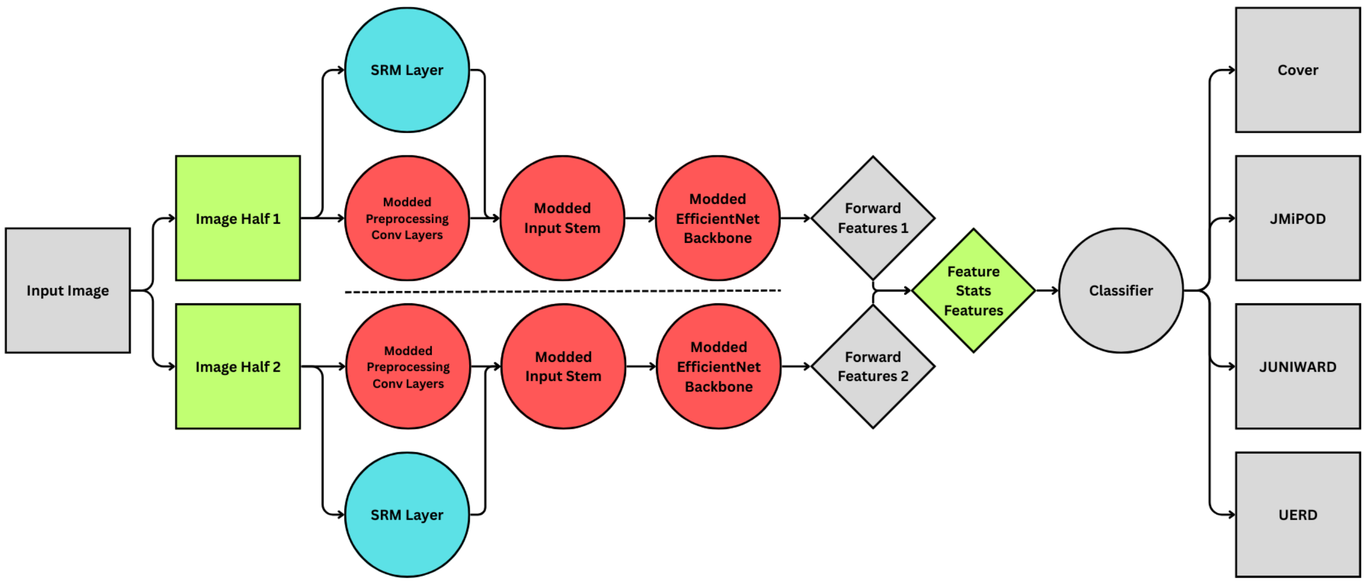

Figure 2: An illustration of the entire EfficientNet pipeline with all three modifications applied. The items in green were introduced with the Siamese modification, blue with high-pass filtering, and red with the architectural modification to emphasize lower-level features. Note that the architecture blocks below and above the dotted line have the same weights.

channel attention mechanism through Squeeze-and-Excitation (SE) blocks [8], allowing the model to focus on important regional features while suppressing irrelevant information. The architecture consists of a noise extraction module, noise analysis module, and classification module, each optimized to enhance discriminative learning in complex textured regions.

SiaStegNet adopts a Siamese network architecture combined with a high-pass SRM (Spatial Rich Model) filtering layer to enhance steganalysis performance. The Siamese structure allows the model to compare different image sub-regions, leveraging shared feature extraction parameters to detect subtle steganographic patterns. This approach enables steganalysis across images of varying sizes without requiring retraining, making it a robust solution for detecting hidden messages in diverse datasets.

EfficientNet, a widely used lightweight CNN architecture, has been applied to various image classification tasks, including steganalysis. It utilizes a compound scaling method to balance width, depth, and resolution, improving computational efficiency while maintaining strong performance. However, prior work in steganalysis often applies EfficientNet with minimal architectural modifications or preprocessing enhancements, leaving room for further improvements.

In this work, modifications to EfficientNet, incorporating high-pass filtering and structural enhancements to improve its steganalysis capability, are explorated. Through comparative analysis with CCNet and SiaStegNet, The effectiveness of the proposed approach in detecting steganographic content is demonstrated.

3. Methodology

In this study, EfficientNet was utilized as the modular backbone for deep-learning-based steganalysis. The ALASKA2 dataset serves as the benchmark dataset for both training and evaluation. Three preprocessing strategies aimed at improving steganalysis performance were used: a high-pass filter modification using SRM kernels, a Siamese modification for feature comparison, and architectural modifications to enhance low-level feature emphasis. Each modification is detailed in the following subsections and is illustrated in Figure 2.

3.1. High-Pass Filtering Modification

To enhance the extraction of steganographic artifacts, a high-pass filtering layer was integrated using the Spatial Rich Model (SRM) convolution kernels. High-pass filtering is a technique that emphasizes fine details and noise-like residuals in an image while suppressing low-frequency components such as smooth textures and uniform regions. This is particularly useful for steganalysis, as steganographic embedding often introduces subtle artifacts that are more detectable in the noise domain.

In this implementation, the SRM kernels function as predefined high-pass filters, designed to extract noise residuals by removing smooth image content. The high-pass filtering is performed through a custom convolutional layer (SRMConv2d) that applies a fixed set of SRM kernels, precomputed for steganalysis. The layer processes grayscale-converted inputs and generates 30 feature maps, each corresponding to a different high-pass filter that captures essential noise-based patterns. These SRM-filtered outputs can be used independently or concatenated with additional processed features, depending on other modifications applied.

3.2. Siamese Modification

A Siamese network configuration is introduced to compare feature representations of different image regions. Instead of processing images independently, the model is structured to extract pairwise features and compute a similarity metric.

To achieve this, each input image is split into two halves along the width, resulting in left and right image regions. These regions are processed separately through the feature extraction backbone, generating distinct feature embeddings for each half. Once extracted, the feature embeddings are compared using Euclidean distance, as well as statistical measures such as mean, variance, minimum, and maximum values. These statistics are concatenated and passed to a classifier for final prediction.

The Siamese modification aims to enhance the model's ability to distinguish between steganographic and non-steganographic regions by leveraging relational learning between localized image features.

3.3. EfficientNet Architecture Modification

To further improve steganalysis capabilities, three custom architectural modifications were introduced to EfficientNet.

Preprocessing Convolutional Layers: Additional convolutional layers are incorporated before feeding images into the EfficientNet backbone. These layers refine input features and provide an alternative pathway for learning steganalysis-specific patterns. Each layer is followed by Batch Normalization [9] and ReLU6 [10] activation, with channel dimensions increasing from $3 \rightarrow 6 \rightarrow 12 \rightarrow 36$.

Modified Input Stem: The original EfficientNet input stem is replaced with a version that supports a flexible number of input channels. If SRM filtering is enabled, the input channel count increases to 66 (36 from preprocessing + 30 from SRM). The weights of the new convolution stem are initialized by averaging across existing RGB channels, preserving pre-trained knowledge while adapting to the modified input.

Backbone Modifications: The fifth block of EfficientNet is replaced with an identity layer to reduce redundant computations and focus on early-stage feature extraction. The final convolutional expansion layer is replaced with a 1×1 convolution to align with EfficientNet's classifier expectations, ensuring efficient processing of modified feature maps.

The motivation behind these modifications lies in the importance of low-level features in steganalysis. Unlike conventional image classification tasks, where high-level semantic information is crucial, steganalysis relies on subtle noise patterns that are often present in the lower-level representations of an image. By emphasizing early-stage processing and refining input feature extraction, these modifications enable the model to better capture steganographic artifacts, ultimately improving detection performance while maintaining computational efficiency.

3.4. Dataset

The ALASKA2 dataset, sourced from a Kaggle competition, serves as the primary benchmark for this study, as it was used for both training and evaluation. It consists of 75,000 images, each with a resolution of 512 × 512 pixels in RGB format. The dataset is categorized into four distinct classes: Cover, JMiPOD, JUNIWARD, and UERD. The Cover class represents unmodified images, while the remaining three classes correspond to images that have undergone steganographic embedding using different algorithms.

A key characteristic of the dataset is its structure: images within the same index across the different class folders originate from the same base cover image before steganographic modifications were applied. However, not all images are consistently available across all four classes. To maintain stability in training and evaluation, only images present in all four categories were included in the dataset.

The dataset was processed and loaded using a custom data pipeline. During preprocessing, images were randomly sampled while ensuring class balance. Each image was converted to an RGB format and optionally transformed using data augmentation techniques such as random cropping and horizontal flipping. The dataset loader grouped images by their base cover, ensuring that variations of the same image across steganographic algorithms were available for comparative analysis. This approach allowed for a more structured training process, where the model could directly learn differences introduced by different steganographic techniques.

The ALASKA2 dataset presents a challenging and diverse set of images, captured using over 50 different cameras, ranging from smartphones to high-end professional equipment. This diversity aims to mimic real-world conditions where steganographic detection must be robust across varying imaging devices and processing pipelines. As a result, training on this dataset provides a more generalized and reliable model for a real-world steganalysis performance assessment.

3.5. Training Setup

Due to time constraints, the size of the training set, and Kaggle's 30-hour weekly GPU quota (using a P100 GPU), each model was only trained for 20 to 30 epochs. In the training process, all models—including the baselines CCNet, SiaStegNet, and EfficientNet (with each variation)—are trained using 60,000 samples in total, with 15,000 images per class, resulting in 15,000 unique samples. Since the task involves predicting four distinct classes, categorical cross-entropy loss is employed for each model during training. The AdamW [11] optimizer is used with a weight decay and learning rate of 1×10^{-4}.

Each model follows the standard process of computing loss and performing iterative updates. However, SiaStegNet and EfficientNet (only when modified with a Siamese architecture) introduce an additional contrastive loss [12] term alongside categorical cross-entropy loss. This contrastive loss is computed by measuring the Euclidean distance between the feature embeddings (feats0, feats1) of the pair of image regions and determining whether they belong to the same class. Specifically: If two images are of the same class, their feature representations should be closer together, minimizing their Euclidean distance. Otherwise, if they belong to different classes, their feature representations should be farther apart, with a margin that enforces separation.

The contrastive loss $L_{contrast}$ is computed as follows:

$$L_{contrast} = \mathbb{E}[(1-y)\cdot d^2 + y\cdot \max(0, m-d)^2]$$

where:

- d is the Euclidean distance between feature embeddings (feats0, feats1),
- y is a binary label indicating whether the two samples belong to the same class (0) or different classes (1),
- m is a margin hyperparameter (set to 1.25 in this setup) to enforce a stronger separation between dissimilar samples.

The final loss function used to train the model combines categorical cross-entropy loss (L_{cls}) with contrastive loss, weighted by a scaling factor $\alpha = 0.1$:

$$L_{final} = L_{cls} + \alpha * L_{contrast}$$

For training, SiaStegNet follows the original authors' implementation, while EfficientNet is implemented using the timm library. CCNet, however, had to be implemented from scratch, as no open-source code was available. Additionally, because SiaS tegNet does not come with pre-trained weights, it was trained entirely from scratch.

4. Experiments

To evaluate the performance of the proposed steganalysis models, experiments were conducted on a test set consisting of 1,000 samples, evenly distributed among the four classes: Cover, JMiPOD, JUNIWARD, and UERD.

4.1. Results

Table 1 summarizes the performance of the models across multiple evaluation metrics. The overall accuracy (Acc) represents the average classification accuracy across all four classes, while Cover Acc, JMiPOD Acc, JUNIWARD Acc, and UERD Acc indicate the per-class accuracy. The binary accuracy (Bin Acc) reflects performance in a real-world steganalysis setting, where all stego classes (JMiPOD, JUNIWARD, and UERD) are grouped together against the Cover class, effectively transforming the task into a binary classification problem. Finally, the F1 score, which considers both precision and recall, provides a more balanced measure of classification performance.

Among the baseline models, CCNet exhibits the weakest performance, achieving an overall accuracy of only 25.00% due to unstable training. Notably, it classifies all non-cover images incorrectly, resulting in a perfect 100.00% accuracy for the Cover class but 0.00% accuracy for the three stego classes. Similarly, SiaStegNet, while performing better in binary accuracy (63.00%), still suffers from instability, as evidenced by its low overall accuracy (25.00%) and poor F1 score (0.2111). The model demonstrates slightly better classification of JMiPOD and UERD stego classes compared to CCNet but fails to generalize well across all categories.

EfficientNet, serving as a stronger baseline, significantly outperforms both CCNet and SiaStegNet, achieving an overall accuracy of 52.10% and an F1 score of 0.5184. It demonstrates a more balanced classification across all

Model Name	Bin Acc	Acc	Cover Acc	JMiPOD Acc	JUNIWARD Acc	UERD Acc	F1 Score
CCNet	25.00%	25.00%	100.00%	0.00%	0.00%	0.00%	0.1000
SiaStegNet	63.00%	25.00%	24.00%	44.00%	0.00%	32.00%	0.2111
EfficientNet	61.00%	52.10%	**64.00%**	**62.40%**	20.00%	62.00%	0.5184
E-NetSRM (EfficientNet + Mod + HP)	**68.90%**	**55.30%**	53.20%	58.00%	**40.80%**	**69.20%**	**0.5627**

Table 1: Results of baselines compared to the best performing EfficientNet variation, E-NetSRM.

stego classes, with particularly strong performance in distinguishing Cover and UERD images. The binary accuracy of 61.00% further suggests that EfficientNet is capable of differentiating between cover and stego images more effectively than the CCNet and SiaStegNet implementations.

The best-performing model, that is personally deemed "E-NetSRM", which is EfficientNet with both the architectural modification and SRM layer, further improves classification performance, achieving the highest binary accuracy (68.90%) and overall accuracy (55.30%). The per-class accuracies indicate that this model achieves substantial improvements in detecting JUNIWARD stego images, which were particularly challenging for the baseline models. The F1 score of 0.5627 reflects a more balanced classification capability, suggesting that the architectural modifications and high-pass filtering from the SRM layer enhance the model's ability to extract steganographic artifacts.

The observed improvements can be attributed to multiple factors. The architectural modifications, which include additional convolutional layers and alterations to EfficientNet's backbone, likely allow the model to capture fine-grained noise residuals that are indicative of steganographic embedding. The incorporation of the SRM layer further enhances feature extraction by emphasizing high-frequency components, which are crucial for detecting subtle stego perturbations. In contrast, the Siamese modification, while theoretically beneficial for relational learning, appears to introduce instability into training, similar to CCNet, which may explain its inconsistent performance.

These results suggest that while baseline models provide a strong starting point, domain-specific architectural enhancements, particularly those that emphasize low-level feature extraction, are necessary to improve steganalysis performance. The significant increase in both binary accuracy and F1 score when using E-NetSRM indicates that the proposed architectural modifications are effective in distinguishing stego images from cover images, making them more suitable for practical steganalysis applications.

4.2. Ablation Study

To further analyze the contributions of different architectural modifications, an ablation study was conducted using EfficientNet as the base model, evaluating various combinations of enhancements, including high-pass filtering (HP), architectural modifications (Mod), and the Siamese modification (Sia). Table 2 presents the performance of these variants, highlighting the impact of each modification on binary accuracy, overall accuracy, per-class accuracy, and the F1 score.

The baseline EfficientNet model achieves a binary accuracy of 61.00% and an overall accuracy of 52.10%, serving as a strong foundation for further improvements. Introducing high-pass filtering (EfficientNet + HP) improves binary accuracy to 66.00% and increases the F1 score to 0.5547, suggesting that emphasizing high-frequency stego artifacts enhances classification performance. Similarly, incorporating architectural modifications (EfficientNet + Mod) also leads to a performance gain, with an accuracy of 55.40% and an F1 score of 0.5543, demonstrating the effectiveness of structural adjustments in feature extraction. In general, introducing the architectural modifications and/or an SRM layer for high-pass filtering usually results in a consistent accuracy increase.

The best-performing model, EfficientNet with both high-pass filtering and architectural modifications (E-NetSRM), achieves the highest binary accuracy (68.90%) and maintains strong overall classification performance (55.30%). Notably, it significantly improves JUNIWARD accuracy (40.80%), which was previously a challenging class for the baseline model. These results confirm that combining structural refinements with preprocessing

Model Name	Bin Acc	Acc	Cover Acc	JMiPOD Acc	JUNIWARD Acc	UERD Acc	F1 Score
EfficientNet	61.00%	52.10%	64.00%	62.40%	20.00%	62.00%	0.5184
EfficientNet + HP	66.00%	**55.80%**	**72.00%**	63.20%	26.40%	61.60%	0.5547
EfficientNet + Mod	63.90%	55.40%	70.00%	64.40%	23.60%	63.60%	0.5543
EfficientNet + Mod + HP (E-NetSRM)	**68.90%**	55.30%	53.20%	58.00%	40.80%	**69.20%**	**0.5627**
EfficientNet + Sia	66.90%	35.10%	19.60%	40.40%	31.20%	49.20%	0.3511
EfficientNet + Sia + HP	68.80%	26.70%	12.80%	**69.20%**	9.20%	15.60%	0.2239
EfficientNet + Sia + Mod	52.00%	25.10%	47.20%	0.00%	**52.40%**	0.80%	0.1708
EfficientNet + Sia + Mod + HP	65.40%	24.90%	20.80%	21.60%	10.40%	46.80%	0.2330

Table 2: Results of all constructed model variations compared to EfficientNet, showing the effects of each modification.

enhancements yields a model that is both more robust and better suited for steganalysis tasks.

In contrast, the Siamese modification introduces instability into training, leading to inconsistent performance across variants. EfficientNet + Sia achieves a high binary accuracy (66.90%) but suffers from a substantial drop in overall accuracy (35.10%) and a weak F1 score (0.3511), indicating poor generalization across all classes. The addition of high-pass filtering (EfficientNet + Sia + HP) amplifies this instability, with accuracy plummeting to 26.70% despite a competitive binary accuracy of 68.80%. Further integrating architectural modifications (EfficientNet + Sia + Mod) results in even poorer classification performance, with an overall accuracy of 25.10% and an F1 score of 0.1708, similar to CCNet's failure pattern.

Overall, the ablation study highlights the importance of architectural and preprocessing enhancements while also demonstrating the challenges associated with the Siamese approach. The superior performance of E-NetSRM confirms that a combination of architectural refinement and frequency-domain filtering is the most effective strategy for improving steganalysis accuracy.

5. Conclusion

This study explored the effectiveness of preprocessing and architectural modifications in enhancing EfficientNet's performance for steganalysis on the ALASKA2 dataset. By incorporating high-pass filtering using SRM kernels and structural refinements, E-NetSRM was developed, which achieved superior binary accuracy (68.90%) and an improved F1 score (0.5627), outperforming baseline models such as CCNet and SiaStegNet. The ablation study further confirmed the significance of preprocessing enhancements and structural adjustments in improving steganographic detection.

Despite these promising results, there are several limitations to the approach. One major constraint is the instability observed during the training of CCNet and SiaStegNet, which likely stemmed from implementation issues rather than fundamental model design flaws. The inability to stabilize these models suggests that improvements in implementation, hyperparameter tuning, or architectural refinements could yield better results. Additionally, the scope of preprocessing modifications explored in this work was limited to three techniques, namely high-pass filtering, architectural input modifications, and Siamese-based relational learning. However, numerous other preprocessing strategies could further enhance performance. In addition, these models were trained on 30 epochs or less, but it is possible that some variations have not converged.

Future research directions include expanding the range of preprocessing techniques, such as incorporating attention mechanisms alongside high-pass filtering to improve feature extraction. Additionally, refining training methodologies could lead to more reliable results. Exploring alternative architectures that integrate both spatial and frequency domain analysis may also provide valuable improvements in steganalysis performance. Ultimately, the continued development of robust steganalysis methods is crucial in countering evolving steganographic threats and ensuring the security of digital communication.

References

[1] Mingxing Tan, & Quoc V. Le. (2020). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks.

[2] Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, & Edouard Grave. (2019). CCNet: Extracting High Quality Monolingual Datasets from Web Crawl Data.

[3] You, W., Zhang, H., & Zhao, X. (2021). A Siamese CNN for Image Steganalysis. *IEEE Transactions on Information Forensics and Security, 16, 291-306.*

[4] Cogranne, R., Giboulot, E., & Bas, P. (2020). ALASKA#2: Challenging Academic Research on Steganalysis with Realistic Images. In *2020 IEEE International Workshop on Information Forensics and Security (WIFS)* (pp. 1-5).

[5] Cogranne, R., Giboulot, Q., & Bas, P. (2020). Steganography by Minimizing Statistical Detectability: The cases of JPEG and Color Images.. In *ACM Information Hiding and MultiMedia Security (IH&MMSec)*.

[6] Cogranne, R., Giboulot, Q., & Bas, P. (2020). Steganography by Minimizing Statistical Detectability: The cases of JPEG and Color Images. In *Proceedings of the 2020 ACM Workshop on Information Hiding and Multimedia Security* (pp. 161–167). Association for Computing Machinery.

[7] Guo, L., Ni, J., Su, W., Tang, C., & Shi, Y.Q. (2015). Using Statistical Image Model for JPEG Steganography: Uniform Embedding Revisited. *IEEE Transactions on Information Forensics and Security, 10(12), 2669-2680.*

[8] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, & Enhua Wu. (2019). Squeeze-and-Excitation Networks.

[9] Sergey Ioffe, & Christian Szegedy. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift.

[10] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, & Hartwig Adam. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications.

[11] Ilya Loshchilov, & Frank Hutter. (2019). Decoupled Weight Decay Regularization.

[12] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, & Dilip Krishnan. (2021). Supervised Contrastive Learning.