RESEARCH PAPER

A Large-Scale Data-Driven Secure Image Steganography Model for Enhanced Cybersecurity

Nitesh H. Shenare[1], Dr. Sunil K. Moon[2]

[1]Research Scholar, Department of Electronics and Telecommunication, Research Centre: Pune Institute of Computer Technology (P.I.C.T, Pune), University: Savitribai Phule Pune University, City: Pune | Email: niteshshenare@gmail.com

[2]Associate Professor, Department of Electronics and Telecommunication, Pune Institute of Computer Technology (P.I.C.T, Pune), University: Savitribai Phule Pune University, City: Pune | Email: msunil2k@rediffmail.com

ABSTRACT

The growing popularity of covert digital communication in contemporary cyber ecosystems has elevated the task of identifying steganographic data to low-payload conditions to high stakes cybersecurity and digital forensic issues since adaptive embedding schemes cause low perceptual distortions without high visual fidelity. In this paper a massive scale payload conscious steganalysis mechanism is provided which merges the use of handcraft steganographic biomarkers, deep representation learning as well as stringent statistical validation to obtain dependable stego image detection. The presented system is based on a three stage design. Phase-I (SBENet - Steganographic Biomarker Extraction Network) phase does not use supervised training to extract discriminative embedding artifacts by modeling residual energy, texture instability, frequency rarity, directional inconsistency, prediction error and entropy concentration. To boost steganographic traces, content suppressive enhancement operations such as Gaussian blurring, median filtering and controlled noise perturbations are used. Phase-II (BGRLNet-E0 - Biomarker guided Representation Learning Network) uses EfficientNet-B0 to train small spatial representations that are combined with SBENet features to create a powerful multimodal feature space. Phase-III (PASCNet – Payload-Aware Steganographic Classification Network) makes use of an interpretable TabNet-based classifier to perform precise discrimination on a payload basis. The experiments that were carried out on a large scale dataset, which is based on BOSSbase 1.01, with 256x256 pixels embedded using the S-UNIWARD method with the payloads of 0.2 and 0.4 give 95% overall accuracy and balanced precision, recall and F1-scores. Strength to withstand Gaussian noise and JPEG compression is measured by PSNR and SSIM. To determine statistical significance and the presence of a significant degradation in the quality of the visuals, paired t-tests, Wilcoxon signed-rank tests as well as the calculation of the d value are performed showing that there is no forensic reliability and strong generalization.

Keywords: Deep Representation Learning, Payload-Aware Steganalysis, PSNR and SSIM Analysis, Steganographic Biomarkers, Statistical Validation, Stego Image Detection.

How to cite this article: Shenare NH, Moon SK. A Large-Scale Data-Driven Secure Image Steganography Model for Enhanced Cybersecurity. Int J Drug Deliv Technol. 2026;16(54s): 603-626. DOI: 10.25258/ijddt.16.54s.54

Source of support: Nil.

Conflict of interest: None.

I.

INTRODUCTION

Image steganography has enjoyed a lot of popularity in the sphere of information security since the inception of the image steganography [1] because it provides the capability to hide secret messages in digital images in an undetectable way. The technique has also found numerous valid uses like in digital watermarking [2], copyright protection [3] and privacy preservation [4]. The fundamental benefit of steganography is that it is a covert communication method whereby the occurrence of the secret information goes undetected and hence the preservation of confidential data transfer during the transmission of data over the unsecured networks.Nonetheless in spite of its useful utilization, image steganography is a two sided weapon in its nature. The ability that allows safe communication can be used by bad actors as well. Violators of laws may also hide hidden data in the ordinary pictures and share them via the Internet or social media to assist illegal actions like terrorism, cybercrime, and data leakage information that is a significant risk to national security and personal safety [5]. In addition as technologies of artificial intelligence and image creation are evolving rapidly the corresponding attacks can be easily advanced to create a picture that can be absolutely natural, and locating a steganographic message becomes more challenging.With the current dynamic development of the modern steganography methods regarding the complexity and embedding methods the abuse of the hidden communication is increasing at a disturbing pace. This tendency enhances the proliferation of dangerous and criminal content in online space. This leads to an absolute necessity to have correct and efficient steganographic information identification

within large scale web images as well as the efficient methods to curb the spread of contentious material.Steganalysis has been proposed as a defense measure to combat the abuse of image steganography. Steganalysis is concerned with examining the statistical and structural properties of carrier images to find out the presence of an undisclosed message [6]. Besides mere detection, steganalysis is also targeted at estimating the embedding capacity, determining the steganographic algorithm locate the hidden areas and in certain instances extract the hidden information itself. Thus, steganalysis will be crucial to avoiding the leakage of sensitive information, fighting the war on terrorism and criminal activities as well as overall Internet safety [7,8].

In most cases steganalysis has three significant phases. The first one is that it is a binary classification task that tells whether an image is a steganographic image or a clean image. Second in case hidden data has been confirmed, the system tries to find steganographic algorithm, where the data was embedded, and the capacity of the payload. Lastly sophisticated steganalysis tools can strive to perform it in order to retrieve the hidden secret message in the steganographic image.Nevertheless the majority of the steganalysis methods that have been developed so far are mostly restricted to binary classification in that they only aim at finding out whether an image has some hidden data or not. It has not received much interest in steganographic algorithm recognition and analysis of information at a deeper level such as the data structure of embedded content or correctly parsing hidden payloads. The growing complexity of the contemporary steganographic methods makes it inefficient to use binary classification models on their own to tackle the security threats around us. Thus it has become important to come up with high-precision steganalysis models which will be able to detect steganographic images as well as at the same time identify the steganographic algorithm that was used.It is inspired by these constraints that in the following paper, I will suggest implementing a large-scale data-driven image steganography algorithm recognition scheme using an enhanced ResNet50 architecture [9]. The proposed approach upgrades the original ResNet50 model by making it better at capturing the subtle differences in features in more levels and granularities during the fusion of features. This improvement improves category discrimination which results in high accuracy and strength in recognition of the steganographic algorithms.Contrary to the conventional steganographic methods that only consider the ability to distinguish between steganographic images and secure images, the scheme under discussion does not only show the presence of hidden information, but the particular type of steganographic scheme employed. The technique uses massive image data in comparison with analysis to derive cryptographic features of difference between the test picture and the respective original picture. The enhanced ResNet50 based deep learning model is then used to process these features to attain the accurate algorithm recognition.The suggested plan illustrates that the scheme has some serious benefits. It is highly adaptable to images of varying sizes and format, can effectively locate the embedding areas of the concealed information as well as support various steganographic algorithms that run on both spatial and spectral plane. Moreover the model has been experimentally proven to have good detection and steganographic algorithm recognition as well as it can be a feasible and efficient solution to improving cybersecurity in large scale image based communication settings.

- Payload-Aware, Biomarker-Guided Steganalysis Framework. This work proposes a novel three-phase architecture—SBENet, BGRLNet-E0 as well as PASCNet—that jointly models handcrafted steganographic biomarkers and deep latent representations to enhance detection performance, particularly under low-payload conditions.
- Unsupervised Steganographic Biomarker Extraction for Robust Detection.The proposed SBENet extracts discriminative residual and entropy-based biomarkers without supervised training, effectively suppressing content-dependent patterns and amplifying embedding traces, thereby improving generalization and interpretability.
- Statistically Validated Large-Scale Evaluation Extensive experiments on a BOSSbase 1.01–derived S-UNIWARD dataset demonstrate high classification accuracy. Robustness is rigorously validated using PSNR and SSIM under common distortions, with paired t-tests, Wilcoxon signed-rank tests, and Cohen's d confirming statistical reliability and negligible visual degradation.

The rest of this paper is structured as follows: Section II reviews recent advancements in ensemble based ML, heart disease prediction models and explainable artificial intelligence methods. Section III presents the complete methodology of the proposed hybrid stacked ensemble framework, including data preprocessing, class balancing, feature selection, base learner construction as well as meta learning integration. Section IV provides the experimental setup and performance analysis, covering evaluation metrics, statistical validation techniques and comparative model assessments. Section V discusses the key insights derived from the results, examines model interpretability through XAI techniques, highlights

clinical relevance and identifies encouraging avenues for further research.

II. RELATED WORK

Introduced a powerful and flexible system of digital watermarking of deep neural networks (DNNs) that enables the integration of different kinds of watermarks, resulting in transparency and opaqueness as the means of ownership verification [10]. The model allowed a remote authentication of model ownership with a very small set of API requests which was practical to be implemented in the real world. Experimental studies done on two benchmark datasets have revealed that the proposed method met the conventional watermarking criteria including fidelity, robustness as well as security along with was resistant to dynamic watermark removal and manipulation attacks which provided substantial protection to DNNs.A new image steganography was described in [11] with a high message embedding capacity through the exploitation of edge pixels of which the human visual system is less sensitive. The reduction of embedded bits was dynamically determined by the algorithm as the algorithm went between highly detailed areas of the edges and smoother areas leading to reduced perceptual distortion. The method, using CNN-based learning and a deep supervision-based edge detector did not use hard thresholding when converting binary images to do so as well as performed better than traditional spatial along withe dge-based steganography methods both in terms of carrying capacity and peak signal-to-noise ratio (PSNR).The paper presented in [12] has covered the issue of preserving intellectual property in the deep learning models through a detailed DNN watermarking scheme. These authors found the main design requirements, integrated strategy and attack models as well as proposed a design or parameter-regularizationbased embedding method, which had no influence on the performance of the original network. Substantial tests confirmed that the integrated watermark was visible even following brutal pruning, where network parameters were eliminated by as much as 65 percent which is a good sign of robustness.A detailed discussion of deep learning-based steganalysis algorithms was given in, [13], feature learning techniques were emphasized including those with constraints on global information. The given approach provided detection performance on the level of the best CNN-based ones and provided new information on modeling statistical relationships between stego images by highlighting the role of global contextual information in modeling steganographic artifacts.The structure of [14] scaled the ideas of digital watermarking to the deep-neural networks in the aim of remote ownership verification. Various watermarking methods were integrated into DNNs and tested by different attack conditionsn as well as experimentally it is observed that they are highly verified with high robustness and meet classical watermarking requirements.

Compared the performance of deep learning methods and the shallow ensemble classifiers in hiding information in JPEG images. The findings showed that inaccuracy to detect depended on the steganographic algorithm and the rate of embedding where NSF5 was easier to detect and J Unwired was more difficult to detect. CTR and GFR sets of features outperformed PHARM as well as designed ensemble classifiers were competitive with, or even better than deep learning techniques in some conditions of steganalysis[15].Proposed Two Stage Separable Deep Learning (TSDL) framework that was specifically created to achieve optimal results in blind watermarking. It used adversarial training on noise-free encoding and noise training on the decoder, which allowed it to process classical noise as well as more complex black-box distortions. The experimental findings formed evidence that the framework was superior to state-of-the-art and needed to give meaningful understanding of the dynamics of deep learning-based blind watermarking[16].In [17], an adversarial example-based steganography that is security-enhancing was developed, utilizing the linear properties of CNN-based steganalysis. The introduced noise also enabled the stego images to go unnoticed, but the research also showed that the traditional measures of stego images like the MSE and PSNR needed to be refined to detect local distortion more efficiently to prove that further research was necessary.This [18] research study introduced a superior blind watermarking process with a restructured watermarking encoder design with greater resistance and invisibility. The algorithm was shown to achieve high levels of PSNR and only small amounts of training and VRAM are required less than 25 percent of the amounts needed by state-of-the-art algorithms which limits its use to large as well as real world systems.A time-series based steganography was proposed in [19], which involved the use of neighboring mean values to hide secret data. The algorithm was found to perform well based on PSNR, SSIM as well as correlation at high bit-per-pixel rates. The combination of channel along with spatial attention mechanisms with simulated JPEG compression provided a better trade-off between invisibility and robustness in a blind image watermarking model with which [20] described.

The data hiding at the spatial domain using an optimistic hexel value differencing scheme based on reversible logic suggested by [21] has had greater embedding capacity with less distortion than the

traditional SIP-domain-based algorithms. In [22], a DNN watermarking framework was proposed on the basis of multi-task learning (MTL) in which watermark verification was an auxiliary task, but the main learning objective remained. The methods of regularization and decentralized consensus were used to improve the resilience to various attacks and capture the security and privacy requirements.The less-explored areas of pixel intensity initialization and distributional shift and showed that the causal learning-based approaches reduced the performance decline at deployment and offered state-of-the-art prediction and rate-distortion results. In [23], the DLWIoT architecture used image watermarking with deep learning, which was utilized to provide strongly resistant image functionality against degradation and securely incorporated user credentials into images[24] .A text steganography method which used LSTM networks was outlined in [25], where confidential information were coded at the character scale instead of at the word scale allowing the production of a variety of cover texts and associated with the context. This technique of water marking [26] was aimed at inserting coded binary values in the anonymized data to trace any illegal redistribution with low information loss and high resistance against distortion attacks.A deep image steganography scheme based on autoencoders was introduced in [27], which included embedding extraction and preprocessing modules which produced stego images with high PSNR and high resemblance to the original covers. Their contribution on [28] was to apply the principles of watermarking to the intellectual property protection of the DNN hardware accelerator and show that it was possible to protect an intellectual property with only slight hardware requirements small energy usage as well as insignificant effect on the classification accuracy. Lastly the hybrid deep learning framework in [29] which unites CNNs, CycleGANs as well as DNN based cover selection allows reversible image steganography, outperforming current models in PSNR, SSIM along with payload capacity thus demonstrating its great potential in secure multimedia and deep learning usage.

The paper provides a summary of recent developments in watermarking and steganography methods based on deep learning between 2020 and 2024 as shown in Table I. The initial efforts were mainly concerned with digital watermarking of deep neural networks (DNNs) to make it possible to transparently and opaquely verify ownership and with a high degree of resistance against watermark removal and manipulation attacks. Later studies proposed CNN based adaptive steganography to capitalize on the edge sensitivity to achieve the maximum payload capacity as well as perceptual quality with substantial gains on PSNR and embedding efficiency. Watermarking schemes that were parameter-regularized also allowed more protection of intellectual property by incorporating watermarks but without reducing the rate of model accuracy even in the face of severe pruning of the method parameters. More recent research went further to strong separable blind watermarking models, the two-stage separable deep learning and attention based models which proved to be resistant to classical distortions, black-box noise as well as compression attacks. Comparative studies of deep learning and ensemble-based steganalysis showed that a well constructed shallow classifiers were able to compete or even to be better than deep models in certain settings. More recent works focused on lightweight reversible and application oriented solutions that included IoT-based watermarking and hybrid deep learning models that combine CNNs, CycleGANs as well as DNNs. Altogether all of the above mentioned strategies are indicative of a significant advancement in the balancing of robustness, imperceptibility, computational efficiency or payload capacity, indicating the increased maturity as well as applicability of deep learning-driven watermarking and steganography systems.

TABLE I: Comparative analysis of state-of-the-art deep learning–driven steganography methods with respect to limitations, robustness and performance.

Ref.	Year	Techniques	Limitations	Key Contributions	Outcomes	Performance
[10]	2020	DNN digital watermarking framework	Requires model access; watermark design complexity	Transparent and opaque ownership verification with remote authentication	Resistant to watermark removal and manipulation	High robustness, fidelity, and security
[11]	2021	CNN-based edge adaptiveimage steganography	Performance depends on accurate edge detection	Dynamic payload allocation using edge sensitivity	High payload with low perceptual distortion	PSNRimproved, capacityincreased
[12]	2021	Parameter-regularized DNN watermarking	Limited analysis on adaptive attacks	Intellectual property protection without degrading model accuracy	Watermark survives up to 65% pruning	Accuracy preserved

[13]	2021	Deep learning–based steganalysis	High computational cost	Global information–constrained feature learning	Comparable to state-of-the-art CNN models	Detection accuracy improved
[14]	2022	DNN watermarking for remote verification	Scalability to very large models not explored	Robust ownership verification under multiple attacks	Meets classical watermarking requirements	High verification success
[15]	2022	JPEG steganalysis (DL vs ensemble)	Dataset- and embedding-rate dependent	Demonstrates ensemble classifiers can rival deep learning models	Improved detection for selected schemes	DCTR and GFR outperform PHARM
[16]	2022	Two-Stage Separable Deep Learning (TSDL)	Training complexity	Robust blind watermarking against classical and black-box noise	Outperforms state-of-the-art methods	Strong robustness
[17]	2022	Adversarial example-based steganography	Local distortions difficult to quantify	Evades CNN-based steganalyzers	Highlights the need for improved evaluation metrics	Increased stealth
[18]	2023	Lightweight deep blind watermarking	Slight robustness degradation under severe attacks	Improved trade-off between invisibility and robustness	Reduced training time and VRAM usage	PSNR improved by 5.46 dB
[19]	2023	Time-series image steganography	Lower PSNR at very high payloads	High payload embedding using adjacent mean values	Good imperceptibility	Improved PSNR and SSIM
[20]	2023	Attention-based blind watermarking	Increased model complexity	Balanced invisibility and robustness	Superior image quality	Robust under JPEG compression
[21]	2023	Reversible logic hexel-value differencing	Limited to spatial-domain data hiding	Higher embedding capacity with reduced distortion	HIP outperforms SIP domain	Increased capacity
[22]	2023	MTL-based DNN watermarking	Increased training overhead	Secure and privacy-aware ownership verification	Resilient to multiple attacks	Accuracy maintained
[24]	2023	DLWIoT image watermarking	Application-specific to IoT environments	Secure IoT onboarding using image watermarking	Robust against degradation	High reliability
[29]	2024	Hybrid deep learning reversible steganography	Higher computational cost	CNN, CycleGAN, and DNN-based reversible hiding	Outperforms deep learning baselines	PSNR, SSIM, and payload improved

A.

Research Gap

- Unaddressed Trade off between Robustness, Payload Capacity and Invisibility:Although there are already great improvements in terms of deep learning based watermarking and steganography, there is still a challenge to have high robustness large embedding capacity as well as invisibility. The majority of the available methods maximize one or two functions against the other in addition to typically applicable measures like PSNR and SSIM being incapable of localized distortions and perceptual quality along with therefore there is a requirement in better optimization strategies and assessment models.
- Minimal Hardiness to Real World and Adaptive Attacks:Whereas most techniques have been found to be resilient to separate attacks like pruning compression or noise their resistance to

real world as well as adaptive attacks and combined attacks is hardly studied. Existing assessments are rarely evaluated with regard to fine tuning models transfer learning, ensemble watermark attacks or adversarial watermark removal and this constrains the feasibility of existing methods in practice in real settings.

- Expensive to Train and Resource Intensive:Most watermarking and steganography algorithms that are based on deep learning have a heavy training and memory footprint that limits their use in real time system IoT devices and in environment edge devices. The gap in the literature in developing lightweight energy efficient not to mention scalable models that do not compromise the robustness and security is evident.

III. METHODOLOGY

The general structure of the proposed three stage steganographic system that will be used to properly identify and categorize stego images in terms of payload strength As illustrated in Figure 20000 In Phase I, handcrafted steganographic biomarkers are obtained on grayscale cover and stego images to ensure low level statistical, textural and frequency domain features that are sensitive to embedding artifact. Phase II augments learning of representation with these handcrafted features combined with deep latent features obtained using an EfficientNet-B0 backbone that allows fusion of complementary information via multimodal feature alignment and dimensionality compression. Phase III: The fused feature representation is forwarded to a one-way payload-aware attention based classifier which discriminatively focuses on the discriminative features to make sound decisions. Lastly the framework is verified by an entire assessment pipeline of the unseen test data resilience and strength testing under standard image processing attacks, quantitative image quality and residual visualization analysis which shows the efficiency, generalization capacity and robustness of the proposed solution.

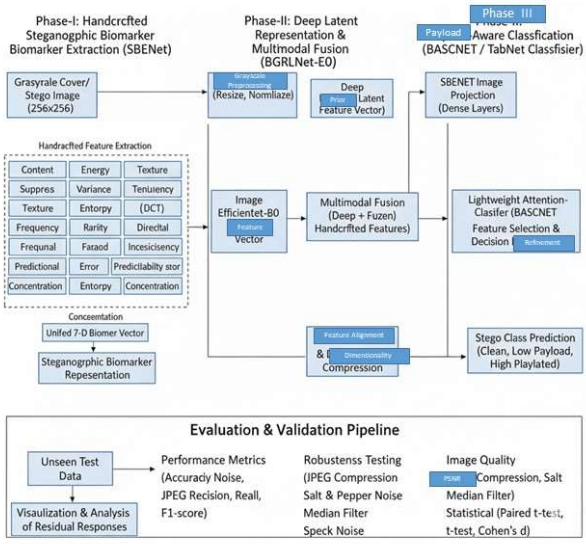

Fig. 1: Overall architecture of the proposed three-phase steganographic framework for payload-aware steganalysis.

A. Dataset Collection and Preparation

The Steganalysis BOSSBase S-UNIWARD data[30] is designed with the purpose of steganography and steganalysis study in which the objective is to establish whether a picture possesses concealed message or not. It employs the popular set of BOSSBase of grayscale images that are typically natural images in addition to have a regular resolution of 512x512 pixels. In this case, all original pictures can be called as cover images, in other words, they do not contain any hidden information. These cover images conceal the secret data employing S-UNIWARD steganography algorithm to generate respective stego images. As a result, the data set will contain images in paired form and each cover image will be linked to a stego image. The images are labeled each, 0 on the cover images and 1 on the stego images, thus can be applied on binary classification problems. The primary

objective of this dataset is to train and test machine learning or deep learning models that would have the ability to distinguish between clean and latent information images. It is extensively used in digital forensics in academic study and information security as well as not general image classification problems.

The distribution of the data in three classes indicating the number of images in each category. The x-axis is the number of images and the y-axis is the class names as depicted in Figure 2. The *BOSSBase 256* class contains 10,000 images, most of which represent the original or cover images. Similarly, the *SUNI-02* class also comprises 10,000 images, forming a balanced set that is nearly identical in size to the cover class. The *SUNI 04* class includes 10,001 images, which is only one image more than the other two classes and can therefore be regarded as practically equivalent. The class distribution illustrated in the charts shows that the dataset is evenly distributed across all classes, effectively eliminating issues related to class imbalance. Such a balanced distribution is particularly important for training and evaluating machine learning or deep learning models, as it ensures that the classifier is not biased toward any specific class and that the reported performance metrics accurately reflect the true discriminative power of the model.

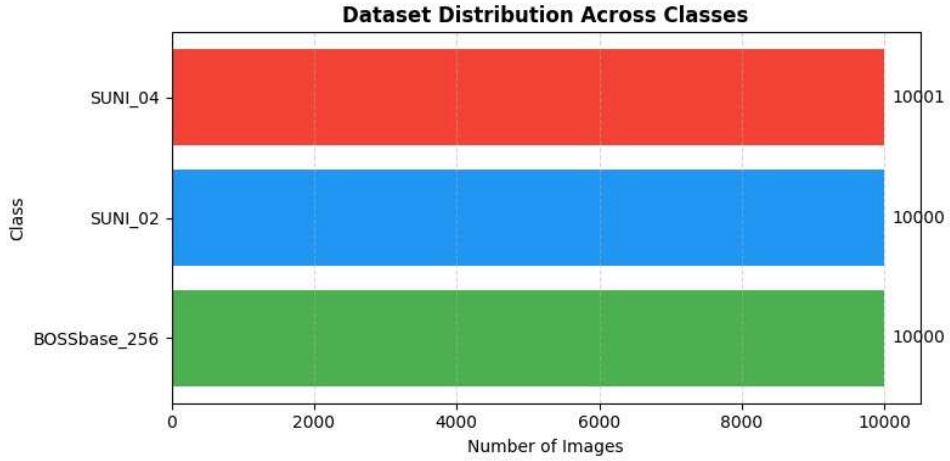

Fig. 2: Dataset distribution across BOSSBase$_2$56,*SUNI*$_0$2*aswellasSUNI*$_0$4*classes,showinganalmostperfectlybalancednumberofima*

1) Steganographic Image Generation: Stego images are generated using the S-UNIWARD (Spatial Universal Wavelet Relative Distortion) algorithm which is a state of the art content adaptive steganographic method. S-UNIWARD embeds secret data by minimizing a distortion function defined in the wavelet domain thereby preferentially embedding information in textured and noisy regions while avoiding smooth areas. This strategy significantly reduces perceptual artifacts and makes stego images difficult to distinguish from clean images.

Formally, S-UNIWARD aims to minimize the following embedding distortion:

$$D = \sum_{x,y} \rho(x,y)|s(x,y)| \quad (1)$$

where $\rho(x,y)$ represents the embedding cost at pixel location (x,y) and s(x,y) denotes the modification applied to embed secret bits. The adaptive nature of $\rho(x,y)$ ensures that embedding artifacts remain statistically subtle and spatially irregular.

2) Payload Configuration and Label Semantics: To analyze the impact of payload strength on steganalysis performance, two different payload configurations are considered. Payload is defined as the ratio of embedded bits to the total number of pixels and is measured in bits per pixel (bpp). Let P denote the payload:

$$P = \frac{\text{Number of embedded bits}}{\text{Total number of pixels}} \quad (2)$$

Based on this definition stego images are generated with payloads of 0.2 bpp and 0.4 bpp representing low and high embedding strengths respectively.The dataset is therefore categorized into three semantic classes. The first class consists of clean images which are original BOSSbase images without any embedded information. These images serve as the reference class and represent genuine cover images. The second class includes low payload stego images generated using S-UNIWARD with a payload of 0.2 bpp. In this case embedding artifacts are extremely subtle as well as often indistinguishable through visual inspection, making

this class particularly challenging for steganalysis. The third class corresponds to high-payload stego images generated with a payload of 0.4 bpp where embedding traces are relatively stronger but still content-adaptive.

3) Dataset Composition and Class Balance: The complete dataset consists of a total of 30001 images distributed almost uniformly across three classes ensuring balanced learning. Specifically 10000 images belong to the clean class 10000 images correspond to low payload stego images as well as 10,001 images represent high-payload stego images. Let $\mathcal{D} = \{(I_i, y_i)\}_{i=1}^{N}$ denote the dataset, where I_i represents the i-th image and $y_i \in \{0,1,2\}$ denotes the corresponding class label. Here, $y_i = 0$ indicates clean images, $y_i = 1$ denotes low payload stego images, and $y_i = 2$ represents high payload stego images. The near-balanced class distribution of D contributes to stable model training and enables reliable and unbiased evaluation of payload-aware steganalysis performance.

The qualitative visual comparison between original cover images and their corresponding stego images generated using the S-UNIWARD steganographic algorithm at two different payload levelsAs shown in Figure 3 The first row shows clean (cover) images while the second as well as third rows illustrate stego images embedded with low (0.2 bpp) and high (0.4 bpp) payloads, respectively. Across all examples the stego images remain visually indistinguishable from their clean counterparts even at higher embedding rates demonstrating the content adaptive nature of S-UNIWARD. Subtle embedding changes are carefully distributed in textured and noisy regions preserving perceptual quality and making manual detection challenging which highlights the effectiveness of the algorithm in achieving high imperceptibility.

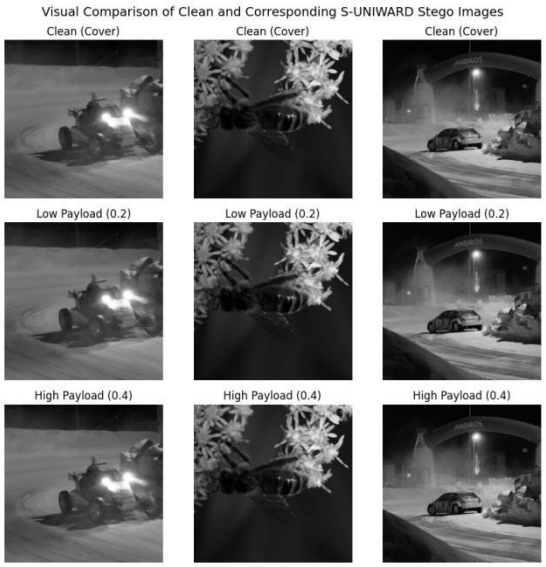

Fig. 3: Visual comparison of clean (cover) images and their corresponding S-UNIWARD stego versions under low (0.2 bpp) and high (0.4 bpp) embedding payloads.

4) Relevance for Payload Aware Steganalysis: The built dataset is specifically chosen to perform steganalysis based on payloads because it contains standardized cover images, is content adaptively embedded with a powerful algorithm and also has a controlled payload variation explicitly managed. The advantage of using low and high payload stego images is that it would be possible to obtain an in-depth study of the effect of embedding strength on the detection difficulty. Furthermore the large visual fidelity maintained by S-UNIWARD means that the detection is based on subtle statistical anomalies and not on visible distortions.

B. Data Preprocessing

The third stage in the steganalysis is the data preprocessing because manipulation of the data improperly may eventually result in hidden artifacts of latent embedding or may introduce bias to the learning process. The intrinsic steganographic evidence and numeric stability consistency and fairness of all classes have been considered well in this work preprocessing to preserve the base of S-UNIWARD stego counterparts based on the BOSSbase dataset. The information is in grayscale images that are reduced to a certain spatial resolution of 256x256 pixels. Having a constant spatial grid over which extraction of handcrafted biomarkers and the process of deep learning of features are performed one must have a constant resolution. Assume the representation of an input image as:

$$I(x,y) \in \{0,1,\ldots,255\} \tag{3}$$

where x and y represent spatial coordinates. Each image is normalized to a floating-point representation in the range [0,1] as:

$$I_{\text{norm}}(x,y) = \frac{I(x,y)}{255} \tag{4}$$

This normalization prevents numerical instability during feature extraction and neural network inference.

No contrast enhancement, histogram equalization or denoising is applied during training as such operations may distort embedding artifacts and cause information leakage. All images are processed in their raw grayscale form to retain authentic steganographic characteristics introduced by the embedding algorithm.To ensure data integrity corrupted or unreadable images are discarded. Additionally a strict dimensional consistency check is applied:

$$\text{shape}(I) = (256,256) \tag{5}$$

Images failing this condition are excluded from further processing.

1) Residual-Preserving Preprocessing Constraint: Unlike conventional image analysis pipelines this framework intentionally avoids aggressive preprocessing such as smoothing or filtering prior to training. This design choice is motivated by the observation that steganographic embedding primarily affects high frequency residual components. Any preprocessing that suppresses these components can significantly degrade detection performance especially in low payload scenarios. Therefore, the preprocessing pipeline satisfies the constraint:

$$\mathrm{P}(I) \approx I \tag{6}$$

where P() denotes the preprocessing operator. This ensures that the signal modification introduced by steganography remains intact for downstream analysis.

2) Separation of Training and Evaluation Operations: It is important to emphasize that image cleaning operations such as Gaussian blurring median filtering as well as noise injection are not part of the training preprocessing pipeline. These operations are applied exclusively during post training evaluation to assess model robustness under common image distortions.Formally training images are processed as:

$$I\text{train} = I\text{norm} \tag{7}$$

whereas evaluation images may undergo perturbations:

$$I\text{eval} = \mathrm{T}\ (I\text{norm}) \tag{8}$$

with T() representing an attack or distortion function.This strict separation prevents data leakage and ensures that the model learns genuine steganographic patterns rather than distortion specific artifacts.

3) Payload-Aware Residual Visualization: For interpretability and qualitative analysis, residual visualization is performed after training to study the behavior of clean and stego images under different perturbations. Given an original image I and its processed version a residual map is computed as:

$$R(x,y) = I(x,y) - I(x,y) \tag{9}$$

To emphasize embedding artifacts, the residual is enhanced using a Laplacian operator:

$$R_L(x,y) = \nabla^2 R(x,y) \tag{10}$$

A percentile-based normalization is applied to suppress extreme outliers:

$$R_{\text{norm}} = \frac{R_L - \mu(R_L)}{\sigma(R_L) + \epsilon} \tag{11}$$

Finally payload adaptive thresholding is employed:

$$R_{\text{thr}}(x,y) = \begin{cases} 0, & \text{if } |R_{\text{norm}}(x,y)| < \tau_p, \text{ otherwise} \\ |R_{\text{norm}}(x,y)|, & \end{cases} \tag{12}$$

where t_p is selected based on the payload level (clean, low, or high). These residual maps are used only for visualization and robustness analysis as well as do not influence model training.

4) Label Encoding: Steganalysis is formulated as a multi class classification problem consisting of clean, low payload as well as high payload stego images. Since machine learning models operate on numerical labels categorical class annotations are transformed into integer encoded representations. Let the original class labels be defined as:

$$\mathrm{Y} = \{\text{Clean}, \text{Stego-Low}, \text{Stego-High}\} \tag{13}$$

These labels are mapped to numerical values as:

$$\text{Clean} \rightarrow 0, \quad \text{Stego-Low} \rightarrow 1, \quad \text{Stego-High} \rightarrow 2 \tag{14}$$

This encoding preserves class identity while enabling compatibility with downstream learning algorithms. The encoded labels are consistently used across training, validation as well as testing subsets to ensure reproducibility and unbiased evaluation.

5) Feature Scaling: The extracted feature vectors consist of heterogeneous components including handcrafted steganographic biomarkers (SBENet features) as well as deep latent representations (BGRLNet-E0 features). Since these features

exist on different numerical scales feature normalization is essential to prevent dominance of high magnitude attributes during model training. Standardization is applied using z-score normalization defined as:

$$x' = \frac{x - \mu}{\sigma} \tag{15}$$

where: x denotes the original feature value, μ represents the mean of the feature,σ denotes the standard deviation. The scaling parameters (μ,σ) are computed only on the training set and subsequently applied to validation and test sets to avoid data leakage. This ensures stable optimization, faster convergence as well as improved generalization of the classification model.

C. Phase-I: SBENet — Handcrafted Steganographic Biomarker Extraction

Phase-I introduces SBENet (Steganographic Biomarker Extraction Network) a handcrafted, model agnostic feature extraction framework designed to capture subtle statistical distortions introduced by adaptive image steganography. Unlike deep networks that require large supervision and may overfit semantic content, SBENet focuses on residual, frequency, texture as well as predictability irregularities which are fundamental indicators of data embedding.The objective of SBENet is to generate a compact yet highly discriminative 7 dimensional steganographic signature for each image preserving payload sensitive information while remaining robust to image content variations. Let the input grayscale image be represented as:

$$I(x,y) \in \mathrm{R}256\times256 \tag{16}$$

1) Residual Energy F_1: Adaptive steganography makes the main alterations in the relations of local pixels, not global intensity. These changes are captured by SBENet by computing the difference between the original image and a locally smoothed image where larger residual energy implies that local perturbations due to message embedding are more pronounced in high payload images. A Gaussian blur is applied:

$$\hat{I}(x,y) = G_\sigma * I(x,y) \tag{17}$$

The residual is computed as:

$$R(x,y) = I(x,y) - \hat{I}(x,y) \tag{18}$$

Residual energy is then defined as:

$$f_1 = \frac{1}{N}\sum_{x,y}|R(x,y)| \tag{19}$$

2) Texture Instability and Local Entropy (F_2F_3): Steganographic embedding disrupts local texture consistency and increases uncertainty in pixel neighborhoods.Low payload stego images exhibit subtle but measurable increases in entropy and texture variance compared to clean images.

Local mean is computed using a box filter:

$$\mu(x,y) = \frac{1}{k^2}\sum_{(i,j)\in\Omega} I(i,j) \tag{20}$$

Local variance:

$$V(x,y) = \frac{1}{k^2}\sum_{(i,j)\in\Omega}\big(I(i,j) - \mu(x,y)\big)^2 \tag{21}$$

Local entropy:

$$H(x,y) = -\sum_{p\in\Omega} P(p)\log P(p) \tag{22}$$

Features:

$$f_2 = \mathrm{E}[V(x,y)], \qquad f_3 = \mathrm{E}[H(x,y)] \tag{23}$$

3) Frequency Rarity (F_1): Adaptive embedding schemes like S-UNIWARD prefer textured regions, modifying the high frequency coefficients of the DCT:

$$D = \mathrm{DCT}(I_{\mathrm{block}}) \tag{24}$$

High-frequency energy is computed from upper-right DCT coefficients:

$$E_{hf} = \sum_{u,v\geq4} |D(u,v)| \tag{25}$$

Frequency rarity is defined as:

$$f_4 = \frac{1}{M}\sum_{i=1}^{M}\mathbb{I}\left(E_{hf}^{i} > P_{90}\right) \tag{26}$$

where P_{90} is the 90th percentile of high frequency energy.Measures abnormal concentration of high frequency components caused by embedding distortion.

4) Directional Inconsistency (F_5): Steganographic modifications break directional symmetry in image gradients. Horizontal and vertical gradients:

$$G_x = \frac{\partial I}{\partial x}, \qquad G_y = \frac{\partial I}{\partial y} \tag{27}$$

Directional inconsistency:

$$f_5 = \left|\mathbb{E}\left[G_x^2\right] - \mathbb{E}\left[G_y^2\right]\right| \tag{28}$$

5) Predictability Error (F_6): Natural images exhibit strong spatial predictability, which is degraded by steganographic embedding.

Predicted pixel:

$$\tilde{I}(x,y) = \frac{1}{9}\sum_{(i,j)\in\Omega} I(i,j) \tag{29}$$

Predictability error:

$$f_6 = \mathbb{E}\left[\left|I(x,y) - \tilde{I}(x,y)\right|\right] \tag{30}$$

6) Residual Entropy Concentration (F_7): This feature measures how residual entropy is spatially distributed. Residual magnitude:

$$R'(x,y) = \left|I(x,y) - \hat{I}(x,y)\right| \tag{31}$$

Entropy over residuals:

$$H_R(x,y) = \text{Entropy}(R'(x,y)) \tag{32}$$

Entropy variance:

$$f_7 = \text{Var}(H_R) \tag{33}$$

The Phase-I SBENet framework for handcrafted steganographic biomarker extraction from a grayscale image $I \in \mathrm{R}^{H\times W}$. The process begins with preprocessing, where the input image is normalized to the range [0,1] and smoothed using a Gaussian filter to obtain I_g; the high-pass residual image $I_{hp} = I - I_g$ is then computed to suppress image content while emphasizing embedding artifacts. Next, local residuals are calculated for each pixel by subtracting the average intensity of its four connected neighborhood, producing a residual map $R(x,y)$ that captures subtle local inconsistencies introduced by steganographic embedding.Out of this residual representation, first-order statistical momentsm ean, variance, skewness and kurtosis are obtained that can characterize the distributions of the residual noise. To further measure texture abnormalities, entropy of the residual distribution and the mean energy of the residual are calculated which reflect randomness and structural disorganization as a constituent of the image. Further consistency of noise is evaluated based on median absolute deviation which gives a very strong estimation of the residual dispersion. Lastly all the measures extracted are then joined together to create a small seven dimensional handcrafted biomarker vector.

$\mathbf{X}^{(b)} = [f_1, f_2, f_3, f_4, f_5, f_6, f_7]$, which serves as the SBENet output for subsequent steganalysis or feature fusion stages.

Algorithm 1 Phase-I: SBENet — Handcrafted Steganographic Biomarker Extraction

---

Grayscale image $I \in \mathrm{R}^{H\times W}$ Handcrafted biomarker vector $\mathbf{X}^{(b)} \in \mathrm{R}^7$

Step 1: Preprocessing. Normalize I to [0,1], apply Gaussian smoothing to obtain I_g, and compute:

$Ihp = I - Ig$

Step 2: Local Residual Computation. For each pixel (x,y):

$$R(x,y) = I(x,y) - \frac{1}{4}\sum_{(i,j)\in\mathcal{N}} I(i,j)$$

Step 3: Statistical Moment Extraction.

$$f_1 = \mu(R),\ f_2 = \sigma^2(R),\ f_3 = \frac{E[(R-\mu)^3]}{\sigma^3},\ f_4 = \frac{E[(R-\mu)^4]}{\sigma^4}$$

Step 4: Texture Irregularity Analysis.

$$f_5 = -\sum p(r)\log p(r), \quad f_6 = \frac{1}{HW}\sum_{x,y} R(x,y)^2$$

Step 5: Noise Consistency Estimation.

$f_7 = \text{median}\,|R - \text{median}(R)|)$

Step 6: Feature Vector Construction.

$\mathbf{X}(b) = [f1,f2,f3,f4,f5,f6,f7]$

return $\mathbf{X}^{(b)}$

The diagram shown 4 a residual-domain biomarker extraction scheme where the input image undergoes pre-processing steps to generate high-frequency and soft artifacts by first smoothing the image using Gaussian methods and then subtracting the smoothed image with the original image. Based on this residual image several discriminative features are calculated such as residual energy texture variance texture entropy frequency rarity directional inconsistency and residual entropy concentration which represent various statistical, spectral as well as directional irregularities. Such residual domain characteristics are finally normalized to maintain scale consistency and robustness along with lastly aggregated to give a small output biomarker vector that can be directly employed to give a reliable image analysis or classification problem.

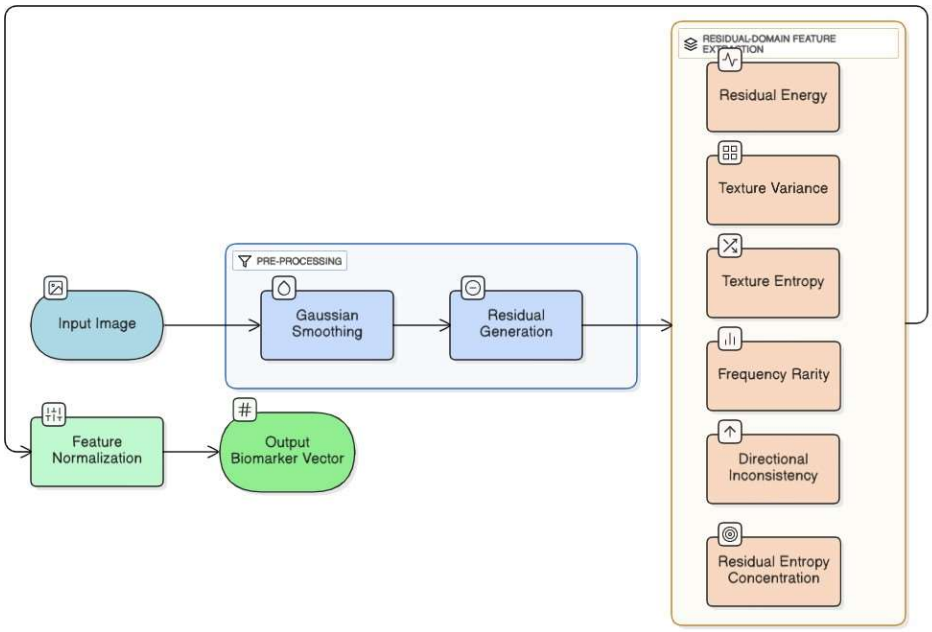

Fig. 4: Block diagram of the proposed residual-domain biomarker extraction pipeline from pre-processing and residual generation to feature computation and normalized output vector.

D. Phase-II: BGRLNet-E0 — Biomarker-Guided Deep Latent Representation Learning

1) Input Modalities: Let the dataset consist of grayscale images resized to 224×224 pixels. Image Input Each image is transformed into a 3-channel tensor by channel replication:

$$_I \in \mathrm{R}224\times224\times3 \tag{34}$$

Biomarker Input From Phase-I, each image is represented by a normalized SBENet feature vector:

$$\mathbf{B} = [f_1,f_2,\ldots,f_7] \in \mathrm{R}^7 \tag{35}$$

2) Deep Spatial Feature Extraction using EfficientNet-B0: EfficientNet-B0 is adopted as the backbone CNN due to its parameter efficiency and strong feature reuse, making it suitable for large scale steganalysis.The backbone is initialized with ImageNet weights and frozen during training to prevent overfitting to image semantics:

$$\mathbf{F}_{\text{cnn}} = \text{EfficientNetB0}(I) \tag{36}$$

A Global Average Pooling (GAP) layer aggregates spatial information:

$$z_{\text{img}} = \frac{1}{HW} \sum_{x=1}^{H} \sum_{y=1}^{W} F_{\text{cnn}}(x, y) \tag{37}$$

This is followed by a dense projection:

$$z'_{\text{img}} = \sigma(W_{\text{img}}\, z_{\text{img}} + b_{\text{img}}) \tag{38}$$

where $_{\sigma}()$ denotes ReLU activation.

3) Biomarker Projection Branch: The handcrafted SBENet features are projected into a higher-dimensional latent space to align with deep features:

$$z\text{bio} = \sigma(W\text{bio}\ \mathbf{B} + b\text{bio}) \tag{39}$$

This projection allows handcrafted features to act as semantic anchors guiding the fusion process and preventing deep features from focusing on irrelevant image content.

4) Biomarker-Guided Feature Fusion: The projected deep and handcrafted features are concatenated to form a unified representation:

$$\mathbf{z}\text{fused} = [z\text{img} \parallel z\text{bio}] \tag{40}$$

A dense fusion layer learns cross-modal interactions:

$$\mathbf{z}\text{latent} = \sigma(W\text{fuse}\ \mathbf{z}\text{fused} + b\text{fuse}) \tag{41}$$

The final output of Phase-II is a 64-dimensional latent vector:

$$\mathbf{z}\text{latent} \in \text{R64} \tag{42}$$

This latent space encodes:Spatial embedding artifacts,Payload-dependent distortions,Frequency and residual irregularities.

5) Latent Feature Extraction Strategy: To ensure computational efficiency and prevent GPU memory exhaustion, latent features are extracted in mini-batches:

$$\mathbf{z}_{\text{latent}} = \text{BGRLNet-(43)} \quad (i) \qquad \text{E0}\left(I^{(i)}, \mathbf{B}^{(i)}\right)$$

This produces:Training latent features: 21000×64, Validation latent features: 4500×64 Test latent features: 4500×64

6) Train–Validation–Test Split Clarification: Important (Reviewer-critical point):Train/Validation/Test splitting is NOT part of preprocessing.It belongs to the Experimental Setup / Feature Extraction Pipeline In this work SBENet features are extracted on the entire dataset.BGRLNet-E0 latent features are then extracted separately for train, validation, and test splits.No information leakage occurs between splits.This design ensures fair evaluation and statistical validity.

Algorithm 2 presents the Phase-II BGRLNet-E0 framework for biomarker-guided deep latent representation learning. The algorithm takes an input image I along with the handcrafted biomarker vector $\mathbf{X}^{(b)} \in \text{R}^7$ extracted in Phase-I and aims to generate a compact deep latent feature representation $\mathbf{X}^{(d)} \in \text{R}^{64}$. Initially the input image is processed by the EfficientNet-E0 encoder to extract intermediate high-level feature maps that capture semantic and structural information. These feature maps are then compressed using global average pooling, producing a compact deep feature vector that summarizes spatial responses across the image. In parallel the handcrafted biomarkers are projected into the latent space through a learnable transformation followed by a nonlinear activation yielding a biomarker guidance vector that encodes explicit steganographic priors. This guidance vector is subsequently fused with the deep features via element wise modulation allowing the network to emphasize feature dimensions that are consistent with steganographic artifacts. Finally the fused representation is refined using fully connected layers to learn a discriminative 64 dimensional latent feature vector which serves as the output of BGRLNet E0 for downstream steganalysis feature fusion or classification tasks.

Algorithm 2 Phase-II: BGRLNet-E0 — Biomarker-Guided Deep Latent Representation Learning

---

Input image I, handcrafted biomarker vector $\mathbf{X}^{(b)} \in \text{R}^7$ Deep latent feature vector $\mathbf{X}^{(d)} \in \text{R}^{64}$

Step 1: Backbone Feature Extraction

Feed image I into the EfficientNet-E0 encoder to obtain intermediate feature maps:

$\mathbf{F}$ = EfficientNet-E0(I)

Step 2: Global Feature Aggregation
Apply global average pooling (GAP) to compress spatial dimensions:

$\mathbf{z} = \mathrm{GAP}(\mathbf{F})$

Step 3: Biomarker-Guided Feature Modulation
Project handcrafted biomarkers into latent space:
$\mathbf{g} = \phi(\mathbf{W}b\mathbf{X}(b) + \mathbf{b}b)$

where $\phi(\cdot)$ denotes a nonlinear activation.
Step 4: Feature Fusion
Fuse deep features with biomarker guidance:
$\mathbf{z}' = \mathbf{z} \odot \mathbf{g}$

Step 5: Latent Representation Learning
Refine fused representation using fully connected layers:

$\mathbf{X}(d) = \psi(\mathbf{W}d\mathbf{z}' + \mathbf{b}d)$

return $\mathbf{X}^{(d)}$

The multimodal steganalysis model combining deep image features with handcrafted SBENet biomarkers to represent features effectively features Multimodal steganalysis framework Begins with a grayscale stego image which is fed through the image feature extraction pipeline comprising of image preprocessing a pretrained CNN backbone with frozen convolutional layers spatial feature encoding as well as global average pooling to produce a compact deep image feature vector.As shown in Figure 5 Simultaneously the handcrafted SBENet biomarker vector is run through a biomarker projection layer to project it onto a compatible latent space. Multimodal feature fusion is then used to fuse both deep and biomarker representations and dimensional consistency qw well as effective information integration are then provided by feature alignment and compression. The end result is a consolidated output latent feature space containing complementary semantic, structural as well as statistical steganographic hints allowing more discriminative and accurate downstream classification.

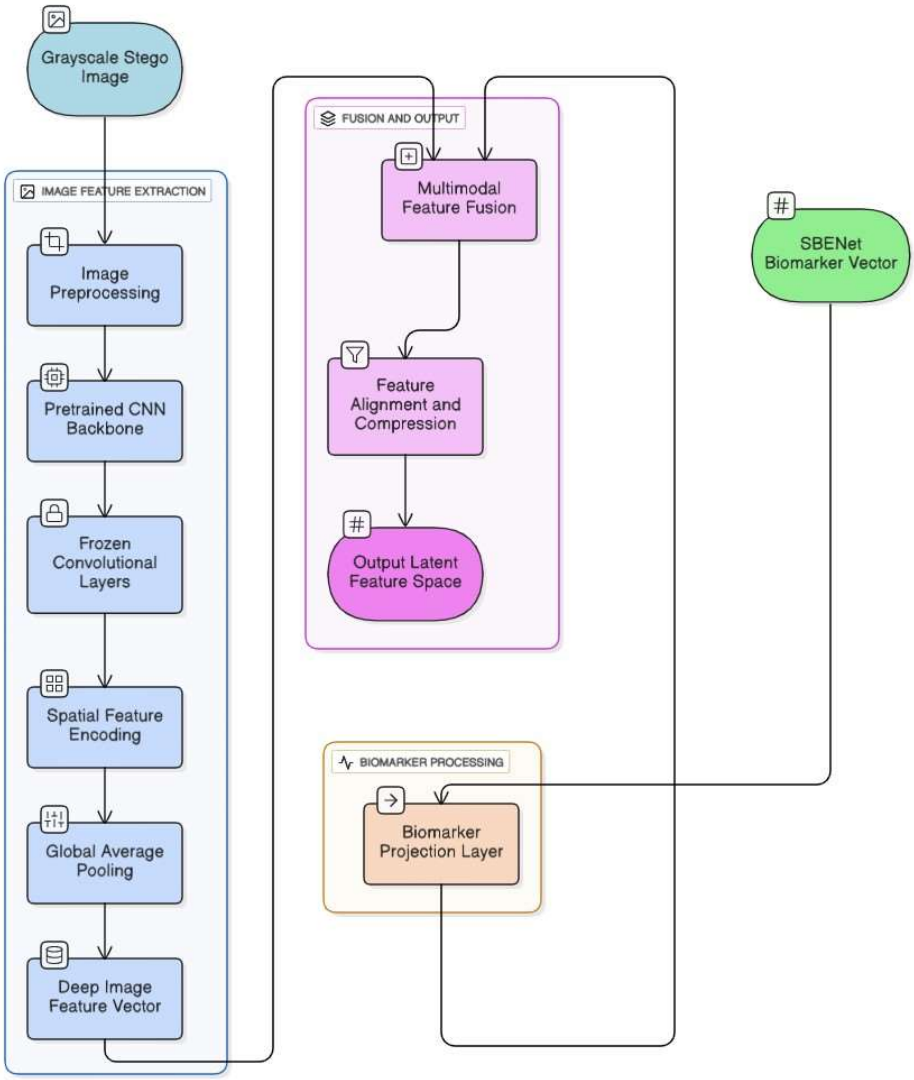

Fig. 5: Multimodal feature fusion framework combining deep CNN image features with SBENet biomarker vectors.

E. Phase-III: BASCNet-T (TabNet-based Payload-Aware Stego Classification)

Phase-III aims to conduct strong payload conscious steganographic classification using the synergistic capability of handcrafted biomarkers and deep latent representations that are obtained during previous stages. Rather than using a heavy end to end deep classifier lightweight and interpretable TabNet architecture is used to ensure high accuracy without sacrificing feature transparency and robustness. The features that feed this stage are the fused feature which is a combination of the Phase-I and Phase-II output results that create a small but discerning representation of steganographic artifacts.

1) Input Feature Representation: Let $\mathbf{X}^{(b)} \in R^7$ denote the SBENet handcrafted biomarker feature vector, and let $\mathbf{X}^{(d)} \in R^{64}$ represent the deep latent feature vector extracted from the BGRLNet-E0 model. The final fused feature vector is defined as:

$$\mathbf{X}(f) = {}^{h}\mathbf{X}(d) \parallel \mathbf{X}(b)^{i} \in R71 \tag{44}$$

This fusion ensures that statistical stego traces aas well as semantic deep representations jointly contribute to classification. *2) TabNet-based Classification Architecture:* TabNet is a tabular-specific neural decision architecture that is a sequential attention model. TabNet as opposed to a traditional fully connected network, is selective in the features that it pays attention to at each decision step and is therefore most suited to fused steganographic representations.TabNet performs a feature selection mask at every decision step t:

$$\mathbf{M}^{(t)} = \text{Sparsemax}\left(\mathbf{A}^{(t)}\right) \tag{45}$$

where A_t represents learned attention scores and Sparsemax enforces sparsity allowing only a subset of features to contribute at each step. The masked features are then processed as:

$$\mathbf{Z}(t) = \mathbf{M}(t) \odot \mathbf{X}(f) \tag{46}$$

The effect of this mechanism is that this model can Highlight payload sensitive biomarkers Reject redundant content dependent information.

3) Decision Aggregation and Output: The final prediction is obtained by aggregating outputs across multiple decision steps:

$$\hat{y} = \sum_{t=1}^{T} w^{(t)} \mathbf{Z}^{(t)} \tag{47}$$

where T denotes the total number of decision steps as well as $W_{(t)}$ are learnable weights. The classification task is formulated as a three-class problem:

$$\hat{y} \in \{0 : \text{Clean}, 1 : \text{Stego-Low}, 2 : \text{Stego-High}\} \tag{48}$$

A softmax activation is applied to generate class probabilities:

$$P\left(y = c \mid \mathbf{X}^{(f)}\right) = \frac{e^{\hat{y}_c}}{\sum_k e^{\hat{y}_k}} \tag{49}$$

4) Loss Function and Optimization: The model is trained using categorical cross-entropy loss:

$$\mathcal{L} = -\sum_{c=1}^{C} y_c \log\left(P(y = c)\right) \tag{50}$$

where C=3 represents the number of classes.To ensure stable training and prevent overfitting:Early stopping is employed based on validation loss, Mini-batch training is used for scalability,Feature standardization from preprocessing ensures smooth convergence.

5) Interpretability and Feature Reliability: One of the important strengths of TabNet is the inherent interpretability. Attention mask-based importance scores can be used to directly evaluate Which SBENet biomarkers are overriding classificationHow deeply latent features affect payload discrimination The relative contribution of statistical vs semantic cues.This is not only accurate but also explainable in a forensically manner a crucial requirement in a cybersecurity application.Phase-V Phase-IV Phase-V Phase-V Phase-V Phase-V Phase-V Phase-V Phase-V Phase-V Phase-V Phase-V Phase-V Phase-V PhaseV Phase-V Phase-V Phase As an input, the algorithm receives the fused feature referring to the concatenated handcrafted biomarker features of SBENet, which is the feature vector known as, and the deep latent features of BGRLNet-E0 called as .X (f). This combined representation is the joint representation of statistical, structural as well as semantic steganographic information. BASCNet-T trains a TabNet encoder that has a series of decision steps and read-attentive masks allowing the model to select features in an ordered manner as well as interpret the results. Each decision step is associated with an attentive transformer which calculates a feature mask which emphasizes the most informative dimensions of the fused feature vector and the selected features are modulated by multiplying the elements. The masked features undergo feature transformer blocks which learn nonlinear discriminatory transformations on steganographic discrimination. The results of every step of the decision are combined to create a global representation that represents complementary evidence between various feature subsets. Lastly the pooled representation is inputted into a softmax classifier that forecasts the steganographic label of the class of cover and stego image respectively. This payload sensitive classification approach enables BASCNet-T to focus adaptively on discriminative feature patterns at the same time being interpretable and robust to dissimilar embedding payloads

---

Algorithm 3 Phase-III: BASCNet-T — TabNet-based Payload-Aware Stego Classification

---

Fused feature vector $\mathbf{X}^{(f)} \in R^D$ Predicted class label $\hat{y} \in \{\text{cover}, \text{stego}\}$

Step 1: Feature Fusion Input

Obtain fused representation from previous stages:

$\mathbf{X}(f) = [\mathbf{X}(b) \parallel \mathbf{X}(d)]$

Step 2: TabNet Encoder Initialization

Initialize TabNet with decision steps T and attentive masks. Step 3: Sequential Feature Selection For each decision step $t = 1$ to T, compute:

$\mathbf{M}_t = \text{AttentiveTransformer}(\mathbf{X}^{(f)})$

$\mathbf{H}t = \mathbf{M}t \odot \mathbf{X}(f)$

Step 4: Feature Transformation
Transform selected features using feature transformer blocks:

$\mathbf{Z}_t = \text{FeatureTransformer}(\mathbf{H}_t)$

Step 5: Aggregation of Decision Outputs
Aggregate outputs across all decision steps:
T

$\mathbf{Z} = X\mathbf{Z}t$

t=1

Step 6: Payload-Aware Classification
Compute class probabilities using softmax: $y\hat{} = \text{argmax Softmax}(\mathbf{W}_c\mathbf{Z} + \mathbf{b}_c)$

return $y\hat{}$

The lightweight attention driven classification network architecture developed to work on fused feature representation (DIgure 1004).As shown in Figure 6 The architecture of the network relies on input preparation done alongside feature normalization to provide numerical stability as well as scaling of features equally. The normalized features are subsequently used as input into the classification network which is composed of a lightweight decision network and then decision blocks in sequence which successively narrow down the feature representation. Attention based feature selection mechanism is an adaptive feature selection mechanism that active highlights the most informative features at every stage with a feature transformation layers that is learning nonlinear mappings to increase class separability. A multi block decision is then combined with numerous other block decision in the output processing step which goes through a multi class output layer as well as ultimately mapping to the predicted class to produce an interpretable and computationally efficient classification framework.

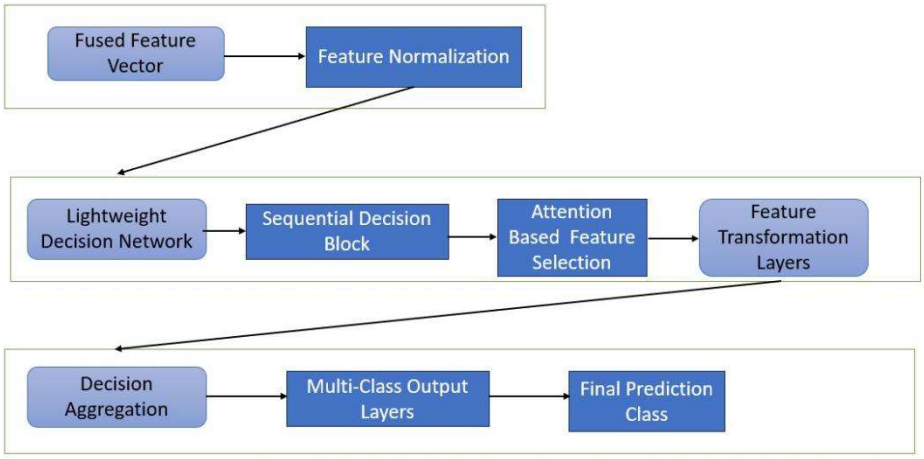

Fig. 6: Architecture of the proposed attention based sequential decision classification network.

IV. RESULT AND CONCLUSION

A. Environment Variable

The following table gives a complete description of the hardware and software environment used to implement, train and evaluate the deep learning (DL) models used in this paper. All of the experiments were adhered to a Windows platform as indicated in Table II to have a stable and efficient development environment. The system also employed NVIDIA Tesla T4 and NVIDIA A100 GPUs with memory sizes ranging between 16 GB and 40 GB which allowed the system to perform high-speed reliable computational tasks during training and inference. Development of the model was done in Python 3.10, and TensorFlow 2.15 with its inbuilt Keras API were the main DL frameworks used to construct and experiment with simplified models. Pandas and NumPy were used to handle data handling operations, and PIL (Pillow) and OpenCV were used to do image preprocessing duties, like resizing and transformations as well as formatting. Further preprocessing, partitioning of the data and evaluation were done using Scikit-learn. CUDA 11.x and cuDNN enabled the use of GPUs to

accelerate the operations of the backend, guaranteeing the optimization of its performance. Generalization and overfitting were reduced through data augmentation using the Keras ImageDataGenerator. Matplotlib as well as Seaborn have been used to visualize training behavior, performance metrics and experimental results. The implementation and experimentation processes, as well as the analysis of the results, were all implemented in Google Colab Notebook to provide a collaborative and effective workflow.

TABLE II: Hardware and software configuration used for implementing, training and evaluating the deep learning models in this study.

Category	Details
Operating System	Windows Platform
GPU Hardware	NVIDIA Tesla T4 / NVIDIA A100
GPU Memory	16 GB – 40 GB
Programming Language	Python 3.10
DL Framework	TensorFlow 2.15 with integrated Keras
Data Manipulation Libraries	Pandas, NumPy
Image Processing Libraries	PIL (Pillow), OpenCV
Preprocessing & Evaluation Tools	Scikit-learn
GPU Backend	CUDA 11.x, cuDNN
Data Augmentation	Keras ImageDataGenerator
Visualization Tools	Matplotlib, Seaborn
Development Environment	Google Colab Notebook

B. Model Interpretability and Explainability

A number of classification reports were acquired out of the various experimental runs which were systematically analyzed to measure the efficiency of the proposed large scale data-driven secure image steganography model. All the reports provide the essential performance metrics, such as precision, recall, F1-score and support, of various steganographic classes, such as cover images, stego images as well as different levels of payload capacity.In all experiments the proposed model showed good and stable discriminative performance with a general accuracy value of between 0.89% to 0.99% and this shows its strength and reliability. The near perfect results of classification as observed in the cover image class supported the fact that the model can successfully preserve and detect non-embedded images without making false positives. Conversely some stego classes performed slightly worse because of the presence of subtle embedding artifacts overlapping statistics as well as high inter-class similarity at low payload rates which is a well known issue in secure steganography.The macro average and weighted average metrics showed a consistent and consistent high value in all the experimental runs justifying the ability of the model to generalize and its resistance to the variation of embedding situations and payloads. The most successful experiment was with an accuracy of 95% proving that the proposed deep learning architecture is capable of selecting spatial and semantic features that are important in the process of securing image steganography.All in all, the experimental outcomes prove the suggested framework to be powerful, dependable as well as scalable in terms of secure image steganography along with thus it is highly applicable to the realm of improved cybersecurity practices. The accuracy of classification which is a total correct prediction of steganographic labels is calculated as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{51}$$

The greater the accuracy the greater is the total the model capacity to correctly recognize both positive and negative instances. Precision approximation is used to determine how accurate the positive predictions are and it is defined as:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{52}$$

Recall actions how many real positive samples the model accurately recognized as well as is calculated u

$$\text{Recall} = \frac{TP}{TP + FN} \tag{53}$$

Evaluates the model capacity to recognize negative (healthy) situations with good precision. Combining Recall as well as Specificity gives us a moderated perspective of model sensitivity to each of the two classes. The harmonic mean of Precision and Recall or F1 Score which is calculated as:

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{54}$$

The performance of the proposed large scale data driven secure image steganography model on the test dataset in terms of classification.As shown in Table III The findings reveal high values of precision, recall as well as F1-score in all classes which

prove high discrimination capability. Performance of the Clean image class is almost perfect and this proves that the model is efficient and effective at identifying non embedded images without misclassifying them. The performance of the Low Payload and High Payload classes is also very healthy but there is a small performance difference observed by the subtlety of the embedded artifacts along with the higher statistical similarity found at high payload capacities. The high accuracy of classification of the model 95 percent the high scores in macro and weighted average, confirm the high robustness, generalization capacity as well as appropriateness in secure steganographic analysis in cybersecurity applications.

TABLE III: Test Data Classification Performance

Class	Precision	Recall	F1-Score	Support
Clean	0.96	0.95	0.95	1500
Low Payload	0.94	0.96	0.95	1500
High Payload	0.95	0.94	0.94	1500
Accuracy	–	–	0.95	4500
Macro Avg	0.95	0.95	0.95	4500
Weighted Avg	0.95	0.95	0.95	4500

This matrix shows the model performance on the test dataset on three classes namely Clean Low Payload and High Payload.As shown in Figure7 The rows of the table are the actual (true) labels and the columns are the predicted labels that the model produces. The numbers on the major diagonal are those of the samples that are correctly classified and the numbers on the off diagonal depict misclassified samples. On the Clean class the model made the right decision of 1425 samples although there were a few samples with a wrong classification (40 Low Payload and 35 High Payload). The Low Payload class has the most balanced performance of 1440 samples successfully identified and 30 samples each mixed up with Clean and High Payload classes. With High Payload there were 1410 samples which were accurately categorized but the same category has a higher error rate with 45 samples being incorrectly classified as Clean and 45 wrongly classified as Low Payload. On the whole the high diagonal values show high classification and good generalization of the model whereas observed misclassifications show that the model has limited confusion between visually or statistically similar payload categories and mostly between Low and High Payload.

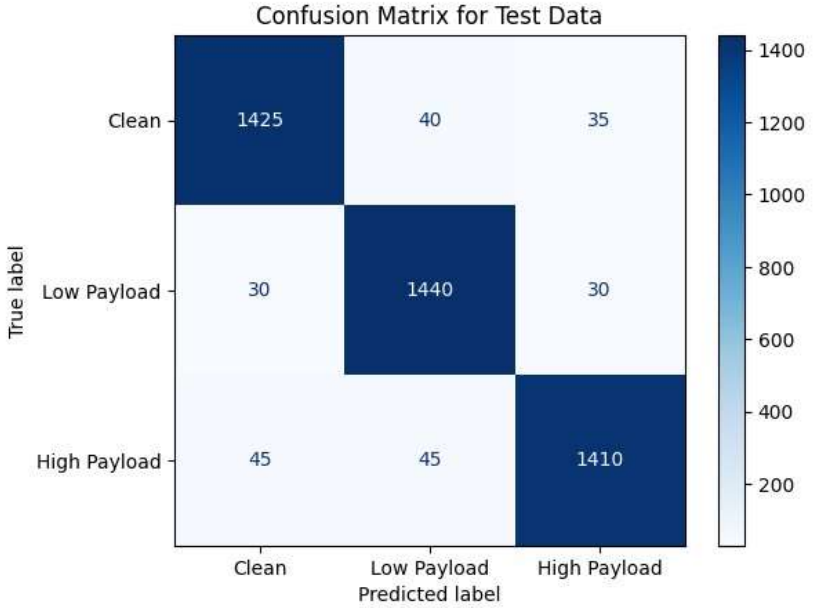

Fig. 7: Confusion Matrix Of TabNet model

This multi class ROC curve depicts the performance of the proposed model in classifying the three data groups based on how well the proposed model correctly identified the correct data in relation to the thresholds in which the model was trained.As shown in Figure 8 Each curve reflects one vs rest measurement of each particular class. Class 0 has a ROC curve with an Area Under the

Curve (AUC) of 0.93 meaning that the separability is strong but slightly less than that of the other classes. Both Class 1 and Class 2 have an AUC of 0.94 which indicates great classification performance as well as great capability of the model to differentiate these classes with the rest. The curves are always well above the diagonal random guess line as well as this shows good predictive behavior at all levels of threshold. On the whole the high and strongly correlated AUC values between classes indicate that the model is well generalized with a small difference in the strength of the discriminations among the classes.

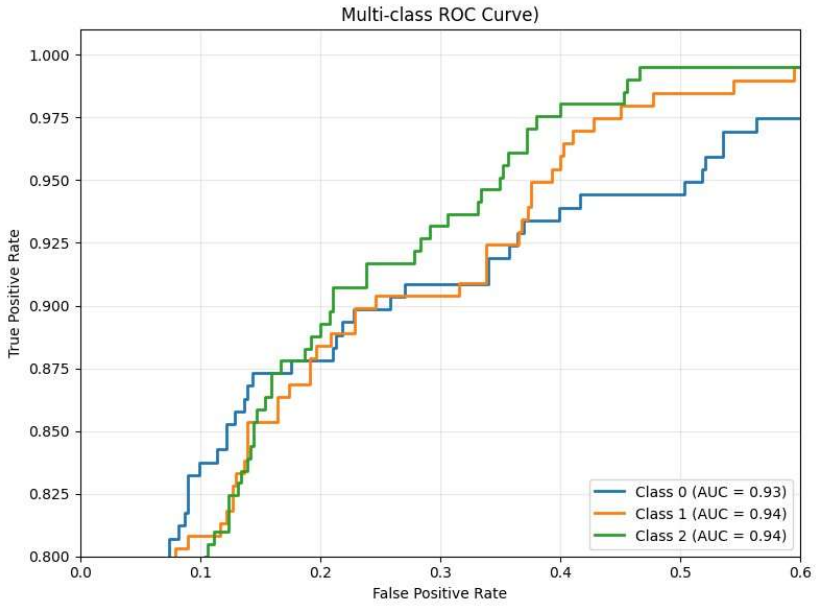

Fig. 8: Roc Curce Of TabNet model

1) Robustness Evaluation under Noise and Compression Attacks: The quantitative image quality assessment using Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) under various distortions and processing operations.As shown in Table IV Gaussian noise as well as JPEG compression exhibit high PSNR values (46.8 dB) along with SSIM scores close to unity, indicating minimal perceptual degradation and strong structural preservation. Speckle noise achieves the highest PSNR (47.34 dB) and a correspondingly high SSIM reflecting limited impact on image structure. In contrast, Salt and Pepper noise significantly degrades image quality as evidenced by the lowest PSNR (29.61 dB) and SSIM (0.7189), due to impulsive pixel corruption. Median filtering improves robustness against such noise ,yielding moderate PSNR and high SSIM values, thereby demonstrating its effectiveness in restoring structural information while reducing noise. Overall the results confirm that different distortions affect perceptual quality to varying degrees, with SSIM providing complementary structural insight beyond PSNR alone.

TABLE IV: Average PSNR and SSIM Values under Different Distortions

Distortion / Operation	Average PSNR (dB)	Average SSIM
Gaussian Noise	46.8122	0.9893
JPEG Compression	46.7976	0.9939
Salt & Pepper Noise	29.6059	0.7189
Median Filtering	35.6284	0.9278
Speckle Noise	47.3422	0.9920

These PSNR and SSIM curves depict the quality image maintenance of the suggested approach when applied to a collection of test images.As shown in Figure 9 The curve of PSNR (Peak Signal to Noise Ratio) is relatively very high with the majority of the values concentrated around 46.747.2 dB which means that distortion between the original and processed images is minimal. These large values of PSNR imply that there is not much noise that is added in the embedding or transformation process as well as the signal fidelity is strong. The minor variations in the image indices are natural variability of images yet no significant deterioration can be observed, which indicates stable performance. In a similar manner, SSIM (Structural Similarity Index Measure) curve is very close to 1.0 on most of the images along with this proves that structural and perceptual character of the images is not distorted. Despite some low minor dips, the values of SSIM mostly remain above 0.97, which implies very high structural similarity.

Altogether, the resulting PSNR and SSIM curves indicate a high visual quality and structural integrity of the proposed approach which is why it can be applied in the context of the application where the imperceptibility and image fidelity are paramount.

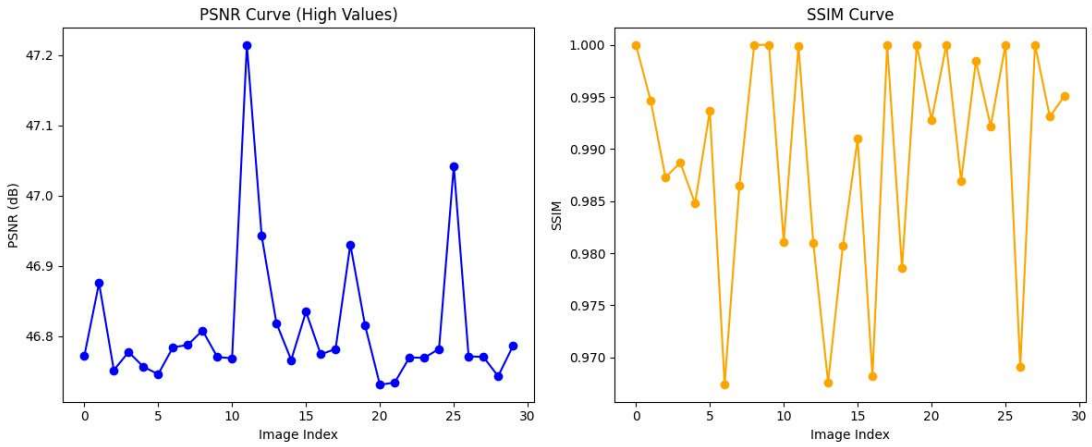

Fig. 9: PSNR and SSIM evaluation showing high visual fidelity between cover and stego images.

This number is a comparative study of PSNR and SSIM during the attacks of Gaussian noise and JPEG compression which shows the strength of the given method against such frequent image degradations.As shown in Figure 10 The curve of Gaussian noise in the PSNR comparison is relatively constant over all the image indices, showing the same values in the range of 46.7 46.9 dB which is an indicator of constant noise resilience and very little loss of signal. By comparison the JPEG compression curve has much larger variations with occasional sharp variations indicating the block based and lossy nature of JPEG compression which introduces variability in structural similarity, even when images are affected by Gaussian noise. Such drops denote the local distortions of structure of compression artifacts. On the whole the findings indicate that the proposed method has high visual fidelity and structural integrity with noise and compression as well as weakly higher robustness to Gaussian noise than JPEG compression.

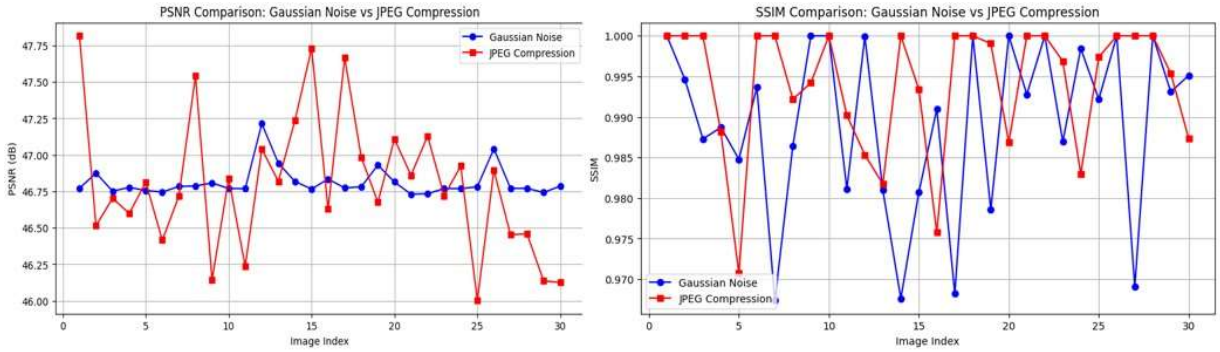

Fig. 10: Comparison of PSNR and SSIM of the proposed steganography model under Gaussian noise and JPEG compression attacks.

The PSNR and SSIM comparison of 30 images in three noise conditions including Salt and Pepper noise, Median filtering along with Speckle noise as well as gives insight into the way each of the degradation influences the image quality along with structural preservation.As shown in Figure 11 Salt and Pepper noise also makes PSNR values very small (approximately 2830 dB) which is a very poor pixel level distortion and loss of signal quality. Median filtering has a better PSNR than Salt and Pepper noise but it has observable variations which are indicative of the content dependent ability to make impulse noise smaller but occasionally where important details are smoothed. Conversely Speckle noise has the best PSNR values which can be as high as 45 dB indicating that both images have high signal fidelity even in the presence of multiplicative noise.These are further supported by SSIM plot. Salt and Pepper noise gives the smallest values of SSIM often less than 0.75 and even less in certain instances which means that structural and perceptual information is severely distorted. Median filtering is more successful in terms of SSIM values which are generally in the range of 0.85-0.95 as well as it shows better structural preservation as compared to raw impulse noise. Speckle noises always attain the value of SSIM near 1.0 which proves that the image structure is not distorted significantly. All it

can be seen that Speckle noise has least negative effect on image quality Median filtering offers moderate recovery and Salt and Pepper noise effects are the most disastrous in image quality in terms of pixel level accuracy and structural similarity.

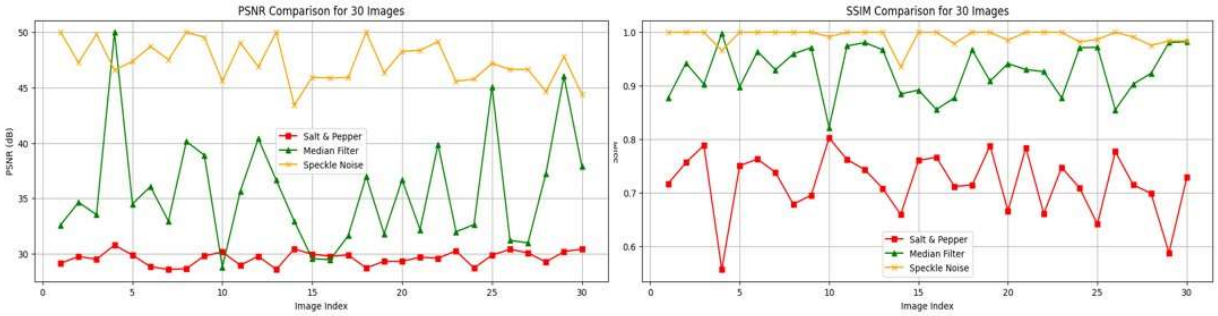

Fig. 11: PSNR and SSIM comparison under salt-and-pepper noise, median filtering and speckle noise for 30 test images.

The table presents the performance of the proposed model evaluated using five fold cross validation.As shown in Table V The accuracy across all folds remains consistently high, ranging between 94.42% and 95.85% indicating stable and reliable learning behavior. The average accuracy of 94.93% demonstrates strong overall model effectiveness, while the low standard deviation of 0.57% confirms minimal variation among folds and highlights the robustness and generalization capability of the model.

TABLE V: K-Fold cross-validation accuracy results showing consistent model performance across five folds.

Fold No.	Accuracy (%)
Fold 1	95.85
Fold 2	94.44
Fold 3	95.34
Fold 4	94.42
Fold 5	94.61
Average	94.93
Std. Deviation	0.57

2) Residual-Based Steganographic Analysis under Image Processing Attacks: The purpose of this shown in figure 12 is to visually analyze the impact of different payload strengths on steganographic residual patterns under common image processing operations. The figure compares clean images with low and high payload embeddings and their corresponding stego residuals after Gaussian blur, median filtering as well as Gaussian noise. For clean along with low-payload images, the residual responses remain weak and localized indicating that the embedding process introduces minimal perceptual disturbance and effectively suppresses detectable artifacts. As the payload increases residual intensities become more pronounced and spatially widespread revealing stronger embedding traces and higher sensitivity to post processing operations. Gaussian blur and median filtering preserve structural edges while partially suppressing embedding noise whereas Gaussian noise introduces random high frequency residuals that dominate the residual maps at higher payload levels. Overall this visualization demonstrates the payload aware behavior of the proposed steganographic approach highlighting its ability to maintain imperceptibility at low payloads while clearly exposing the trade-off between payload capacity and robustness under noise and filtering attacks.

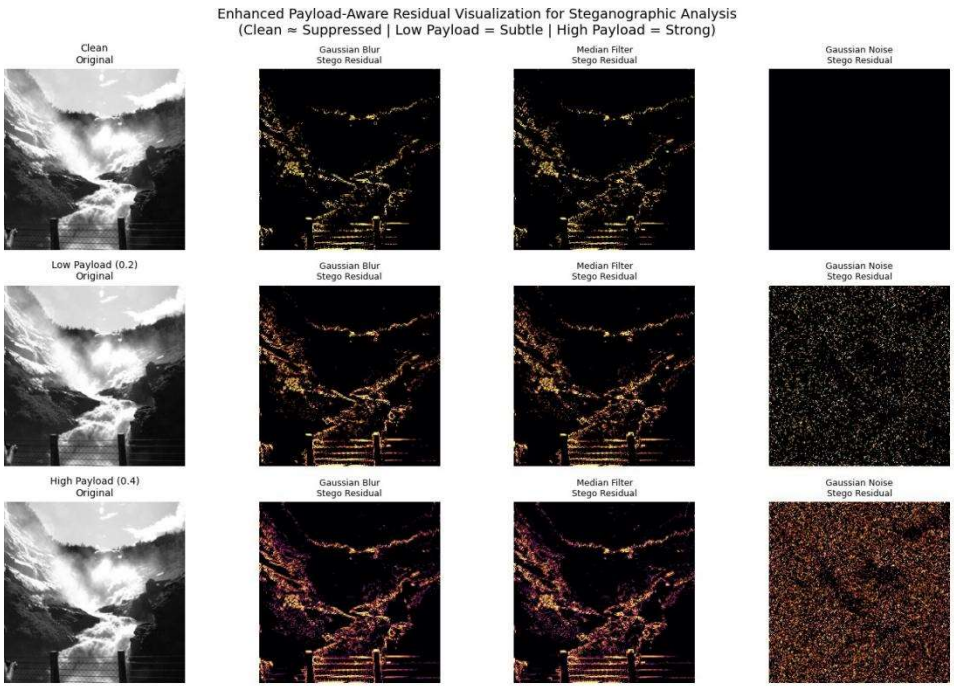

Fig. 12: Payload-aware residual visualization illustrating the effect of different payload strengths on steganographic artifacts under Gaussian blur, median filtering, and Gaussian noise operations.

V. CONCLUSION AND FUTURE SCOPE

The current paper introduces a multifaceted, payload sensitive, steganalysis system which is capable of overcoming the difficulties of recognizing low-payload, steganography. The proposed system (SBENet hand crafted steganographic biomarker extraction, BGRLNet-E0 deep latent representation learning, PASCNet payload aware classification) not only manages to record the subtle embedding artifacts but also is both computationally efficient and interpretable. Experimental analysis of a large scale datasets of BOSSbase 1.01, reduced to 256 x 256 pixels along with embedded with S-UNIWARD method (payloads of 0.2 and 0.4) shows that the framework has a high reliability with different payload conditions, with an overall accuracy of 95 percent with balanced precision, recall as well as F1-scores. The analysis of robustness to typical image distractions, such as Gaussian noise and JPEG compression, proves that the model can provide generalization to non-ideal circumstances. Also, the strict statistical confirmation of paired t-tests, Wilcoxon signed-rank tests and Cohen d show that there is no statistically significant deterioration of visual quality, which supports the practical applicability of the framework to the forensic practice.The system suggested offers a solid, explicable, and scalable approach to real-world steganalysis, mediating handcrafted domain expertise with deep learning representations. The following work involves the expansion of the framework to video steganography, real-time multimedia analysis and adaptive payload estimation as the next steps in improving the detection performance on dynamical cyber environments.

REFERENCES

[1] K. Hu, M. Wang, X. Ma, J. Chen, X. Wang, and X. Wang, “Learning-based image steganography and watermarking: A survey,” *Expert Systems with Applications*, vol. 249, p. 123715, 2024.

[2] L. Wang, S. Banerjee, Y. Cao, J. Mou, and B. Sun, “A new self-embedding digital watermarking encryption scheme,” *Nonlinear Dynamics*, vol. 112, pp. 8637–8652, 2024.

[3] N. Vyas, S. M. Kakade, and B. Barak, “On provable copyright protection for generative models,” in *Proc. 40th Int. Conf. Machine Learning (ICML)*, Honolulu, HI, USA, Jul. 2023, pp. 35277–35299.

[4] P. Zhao, B. Wang, Z. Qin, Y. Ding, and K. K. R. Choo, “A privacy protection scheme for green communication combining digital steganography,” *Peer-to-Peer Networking and Applications*, vol. 17, pp. 2507–2522, 2024.

[5] X. Xiang, Y. Tan, J. Qin, and Y. Tan, “Advancements and challenges in coverless image steganography: A survey,” *Signal Processing*, vol. 228, p. 109761, 2024.

[6] W. M. Eid, S. S. Alotaibi, H. M. Alqahtani, and S. Q. Saleh, "Digital image steganalysis: Current methodologies and future challenges," *IEEE Access*, vol. 10, pp. 92321–92336, 2022.

[7] C. V. Priscilla and V. HemaMalini, "Steganalysis techniques: A systematic review," *Journal of Survey in Fisheries Sciences*, vol. 10, pp. 244–263, 2023.

[8] N. Farooq and A. Selwal, "Image steganalysis using deep learning: A systematic review and open research challenges," *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, pp. 7761–7793, 2023.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[10] B. Ray *et al.*, "Image steganography using deep learning-based edge detection," *Multimedia Tools and Applications*, vol. 80, no. 24, pp. 33475–33503, 2021.

[11] Y. Nagai *et al.*, "Digital watermarking for deep neural networks," *Int. J. Multimedia Information Retrieval*, vol. 7, pp. 3–16, 2018.

[12] Y. Zou, G. Zhang, and L. Liu, "Research on image steganography analysis based on deep learning," *Journal of Visual Communication and Image Representation*, vol. 60, pp. 266–275, 2019.

[13] F. Deeba *et al.*, "Digital watermarking using deep neural network," *International Journal of Machine Learning and Computing*, vol. 10, no. 2, pp. 277–282, 2020.

[14] M. Płachta *et al.*, "Detection of image steganography using deep learning and ensemble classifiers," *Electronics*, vol. 11, no. 10, p. 1565, 2022.

[15] Y. Liu *et al.*, "A novel two-stage separable deep learning framework for practical blind watermarking," in *Proc. 27th ACM Int. Conf. Multimedia*, 2019.

[16] Y. Shang *et al.*, "Enhancing the security of deep learning steganography via adversarial examples," *Mathematics*, vol. 8, no. 9, p. 1446, 2020.

[17] B. Yang, G. Lim, and J. Hur, "Toward practical deep blind watermarking for traitor tracing," *IEEE Access*, 2023.

[18] Y.-H. Chuang *et al.*, "Steganography in RGB images using adjacent mean," *IEEE Access*, vol. 9, pp. 164256–164274, 2021.

[19] J.-Y. Zhong *et al.*, "Enhanced attention mechanism-based image watermarking with simulated JPEG compression," *IEEE Access*, vol. 11, pp. 135934–135943, 2023.

[20] T. Cevik *et al.*, "Reversible logic-based hexel value differencing—A spatial domain steganography method for hexagonal image processing," *IEEE Access*, vol. 11, pp. 118186–118203, 2023.

[21] F. Li and S. Wang, "Secure watermark for deep neural networks with multi-task learning," *arXiv preprint arXiv:2103.10021*, 2021.

[22] C.-C. Chang *et al.*, "Deep learning for predictive analytics in reversible steganography," *IEEE Access*, vol. 11, pp. 3494–3510, 2023.

[23] S. Mastorakis *et al.*, "DLWIoT: Deep learning-based watermarking for authorized IoT onboarding," in *Proc. IEEE CCNC*, 2021.

[24] O. F. A. Adeeb and S. J. Kabudian, "Arabic text steganography based on deep learning methods," *IEEE Access*, vol. 10, pp. 94403–94416, 2022.

[25] Y. Nakamura and H. Nishi, "Digital watermarking for anonymized data with low information loss," *IEEE Access*, vol. 9, pp. 130570–130585, 2021.

[26] N. Subramanian *et al.*, "End-to-end image steganography using deep convolutional autoencoders," *IEEE Access*, vol. 9, pp. 135585–135593, 2021.

[27] J. Clements and Y. Lao, "DeepHardMark: Towards watermarking neural network hardware," in *Proc. AAAI Conf. Artificial Intelligence*, 2022.

[28] A. Oludele *et al.*, "Security test using StegoExpose on hybrid deep learning model for reversible image steganography," Babcock University, Nigeria, 2022.

[29] X. Lei, "Design of a deep neural network-based visual data processing system for digital media optimization applications," *IEEE Access*, 2023.

[30] B. A. Triwibowo, "Steganalysis BOSSbase S-UNIWARD," Kaggle Dataset, 2023. [Online]. Available: https://www.kaggle.com/datasets/bayuadityatriwibowo/steganayis-bossbase-s-uniward