



Enhance Steganography Payload based on Deep Learning Model

Mohammed Mahdi Hashim

Studies, Planning & Follow Up, Directorate, Ministry of Higher, Education and Scientific Research, Baghdad, Iraq

Abstract

Image steganography is the process of hiding confidential information within digital images by introducing subtle modifications that are generally imperceptible to human vision. However, conventional steganographic techniques often encounter challenges related to limited data-hiding capacity, reduced robustness, and vulnerability to various forms of analysis or attack. This paper proposes image steganography based on generative adversarial networks, called IM-GAN, used to improve the capacity of the image steganography framework. The proposed work includes three steps: first, the encoder steps used for embedding secret data into cover images. The second is a decoder for secret message recovery. Thirdly, a critic for visual quality assessment and a noise layer to enhance robustness against attacks. Through the combination of densely connected convolutional features, a Residual Kolmogorov Arnold Network (RKAN)-based convolutional layer, and a multi-objective loss function, IM-GAN enhances hiding capacity while preserving high visual quality and maintaining strong security. Three different datasets are used (DIV2K, MS COCO), along with BossBase 1.01 (512x512 pixels), to evaluate the results. The IM-GAN attains an embedding capacity of up to 4 bits per pixel (bpp) with high decoding accuracy and minimal perceptual distortion, even under Gaussian noise interference. Different parameters used to evaluate the proposed work demonstrate enhancement in terms of Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), Human Visual System (HVS), SRNet, XuNet, and Ye-Net, while still achieving competitive performance even when trained using a relatively small dataset.

Keywords: Deep learning, Steganography, Security, Peak Signal-to-Noise Ratio (PSNR), Generative Adversarial Networks, SRNet, XuNet, Ye-Net.

OPEN ACCESS

ISSN 2541-5107 (online)

*Correspondence:
Abdullah A. Nahi
eng.abdullah.nahi@gmail.com

Citation:
Mohammed
Mahdi Hashim. (2026) Enhance
Steganography Payload based on
Deep Learning Model

Journal of
Information and
Computer
Technology Education.
10.21070/jicte.v10i1.1726

I. Introduction

In recent times, the fast development of information and communication technology has enabled the free and easy transfer of the vast amount of data over the open internet. This free flow of massive data over the internet network in turn posed a severe threat towards the privacy preserved sensitive data transfer where the intruders/attackers are constantly faced. Although the transfer (sending or receiving) of the data information (such as the video, audio, image, and text) became very easy, securing the sensitive information over the insecure public network posed new challenges. For secured information transfer from the sender to the received end, diverse information hiding techniques, including steganography, cryptography, and watermarking, have been introduced. Using the steganography scheme, the secret or private data can be hidden within different media, including the color or grayscale image (Singh et al., 2019).

The steganography technique can be categorized into various types based on the cover media such as the image, audio, text, video, DNA, or even protocol (Hussain et al. 2018) wherein every such cover media has its own merits and demerits (Dhar and Banerjee 2019; Kadhim et al., 2019). Over the past decades, intensive studies have been performed to develop some highly robust and secured image steganography schemes (ISSs). Essentially, these ISSs became promising owing to their easy transmission capacity of the multimedia contents via different low-cost devices (for example smart mobile phones and IP digital cameras) and social media application platforms such as the WhatsApp, Twitter, Facebook and LinkedIn (Hussain et al., 2018). Despite the popularity and robustness of these ISSs, various issues related to the image security and concealment of the secret message for the safe information transfer remains unresolved. In addition, a better understanding of the secret data embedding processes in an image is vital

for the development of an outperforming ISS (Sahu and Swain, 2020).

Modern steganographic techniques typically embed information by modifying a carrier medium[3]. Digital images are the most prevalent carriers due to their widespread availability and high redundancy. Secret information is often encoded into low-level features such as pixel values[4] or Discrete Cosine Transform (DCT) coefficients. Conversely, steganalysis aims to uncover evidence of covert communication by detecting statistical anomalies or feature distortions introduced during the embedding process[5]. This adversarial dynamic constitutes the core conflict driving the evolution of steganography, a struggle that is structurally analogous to the attack-defense paradigm in traditional information security. The steganography technique has been applied in the field of medical diagnoses (Arunkumar et al., 2019; Eze et al., 2019), military and defense (Tuncer and Avci, 2016), multimedia biometric data security (Mohsin et al., 2018) and cloud computing (Shanthakumari and Malliga, 2019). The fundamental issues and difficulties concerning the performance of the existing state-of-the-art steganography schemes are related to the payload capacity, imperceptibility, and security (Hussain et al., 2018).

The payload capacity of a steganography system is defined as the maximum size of the secret message that can be hidden into the image media (Kadhim et al., 2019). The maximum payload refers to the highest limit of the secret data that can be embedded into the image. The imperceptibility of a steganography system signifies the carrier media quality that can be used for hiding the secret message following the algorithm embedment (Hussain et al., 2018). The security of a steganography system (Swain et al., 2019) refers to its robustness against various statistical attacks such as the chi-square, human visual system (HVS) and histogram analysis. In this perception, present study intends to resolve various issues related to the existing steganography system and improve its robustness in terms of high security, high payload capacity, and imperceptibility.

The enhancement of imperceptibility (embedding procedure) was an important subject of various works in steganography and steganalysis, which is significant to note. Based on the literature review, the PSNR and SSIM were used to test the performance of imperceptibility [23][24][12][1]. Indeed, one of the greatest difficulties with the steganography system is improving SSIM and

PSNR. Several studies have been suggested to improve the SSIM and PSNR values, whereas the payload capacity is a Trade-Off between those two metrics. The trade-off between those two issues is the main research gap in image steganography. Nevertheless, despite the achievement of some intriguing findings, only a few studies on the payload capacity have been carried out [27][28][29][30][25][26]. Therefore, the trade-off between the steganography requirements is not balanced. In order to improve the PSNR and SSIM values, many investigators used a small payload capacity [31][14]. According to observations, using a large payload capacity degraded the quality of the image by lowering SSIM or PSNR and vice versa [2][14][30][31][32]. When the PSNR value is equal to 45 dB or below, it is regarded as low imperceptibility [33]. In SSIM, if the result of SSIM equals or is close to 1, that means the two images are considered similar (original and stego image) [3][17][34][35].

Based on the security issues in image steganography, various methods have been extensively debated in recent research [18][19][20]. Various methods to address this issue using noise removal, data encryption, etc., security flaws continue to exist. Consequently, it became necessary to develop new and more effective techniques for concealing confidential data with security [21][22]. Most of the methods used the LSB technique in order to hide confidential data due to the simple embedding within the LSB component of the pixel. These methods have certain limitations in terms of security issues. Different methods have been introduced to improve security. The pixel randomization method was more effective than other methods because it used the same pixel characteristics [23][24].

Based on the above, this explains the absence of studies exploring more than one criterion, thus making comparing previous approaches difficult or impossible, as mentioned in the related work section with different important methods [20][21][22][40]. Therefore,

improving steganography requires a new scheme to solve different issues (imperceptibility, capacity, and security). Evaluation criteria are related to the research gaps. Therefore, the research gaps in steganography might be a challenging multi-criteria issue with competing benchmarking and assessment criteria. To overcome these limitations, the demand for a smarter and more effective method for data hiding that treats security, imperceptibility, and computational complexity is necessitated. The proposed research achieves the following contributions:

- A new convolutional neural network (IM-GAN) is introduced to enhance image steganography performance.
- A novel loss function is introduced to regulate image distortion by adaptively controlling the intensity of the added noise.
- Evaluate the framework based on different metrics (Human Visual System (HVS) attack, Structural Similarity Index Measure (SSIM), Mean Squared Error (MSE), and Peak Signal to Noise Ratio (PSNR)) and security parameters (SRNet, XuNet, and Ye-Net), and comparable performance metrics using a limited number of training samples.

II. Related Work in Steganography and Deep Learning

Image steganography aims to conceal secret information within a cover image while preserving the perceptual quality of the resulting stego image and preventing unauthorized detection of the hidden content. Conventional image steganography generally operates in either the spatial domain or the transform domain. Spatial-domain approaches directly modify pixel values, whereas transform-domain techniques embed information into frequency coefficients obtained through transformations such as the Discrete Cosine Transform (DCT) or Discrete Wavelet Transform (DWT). Although conventional methods are relatively simple and computationally inexpensive, fixed embedding rules can limit their adaptability, payload capacity, and resistance to modern steganalysis. Due to the fact that recent studies have shown how deep learning has become more and more significant as a result of the ability for learned models to simultaneously optimize multiple tasks (feature extraction, embedding, reconstruction, and security objectives), as opposed to solely utilizing hand-crafted embedding methods [1].

Deep learning-based approaches to image steganography have relied on the use of

convolutional neural networks (CNNs) as the primary architecture. CNNs allow for automatic feature learning such as spatial structure, texture, edge, and local statistical properties that are useful in identifying potential locations for embedding secret data. In contrast to traditional techniques that utilize fixed mathematical equations to determine pixel-level or transform coefficient selection criteria, CNN-based image hiding systems extract feature representations from both the cover and secret images. Multi-scale convolution provides an added advantage in terms of providing multi-scale filters that can simultaneously capture detailed texture features and global structural features. As a result, many current state-of-the-art deep steganographic systems employ various types of feature extraction based on multi-scale convolutions; concatenation of extracted features; residual connections; attention mechanisms; and adaptive loss functions to provide better trade-offs between hiding capacity, reconstruction quality, and imperceptibility [30].

An example of a new development in this area has been shown by Yao et al. (2024), who developed an end-to-end image steganography method based on a combination of Generative Adversarial Networks (GANs) and Discrete Wavelet Transforms (DWTs). In their system architecture, they use a multi-scale attention convolutional module to combine the features from different scales, so as to locate where the secret data should be embedded. In addition to locating the best location for embedding the secret data, the system also uses a fusion technique to allow the introduction of the secret data into the image at various levels of modification. The system also includes an adversarial discriminator, which compares the original image (cover) and the modified image (stego), in order to increase the invisibility of the secret data. This research indicates that convolutional neural networks are not only able to extract features from images, but also incorporate attention mechanisms and

frequency domain representations to adaptively determine how to embed secret data into images [31].

The importance of being able to handle various types of image-processing (i.e. JPEG compression, noise, quantization, resize) prior to extracting the hidden message from a stego image is now being recognized as a critical factor in the development of robust deep image hiding techniques. To address this issue, Yang et al. (2024) developed PRIS, a practical robust invertible network for image steganography. PRIS utilizes enhancement modules before and after extraction and uses a multi-stage training process to enhance its resistance to multiple types of distortion. Additionally, the authors addressed the error introduced into the system due to the conversion of neural network output to standard image formats; these errors are typically ignored in purely theoretical implementations. This work illustrates the growing interest in developing deep learning-based steganography models that can maintain their effectiveness after undergoing realistic image-processing operations as opposed to only when they operate in distortion-free environments [32].

Another major area for development is to improve the quality of multi-level feature representations. For example, Cheng et al. (2025) developed MFI-Net, a reversible image hiding system which uses a combination of an up-sampling convolution module and a residual dense module. Both of these types of modules allow the model to use the low-level features (i.e., edges, textures) from the first level of the network as well as the high-level features (i.e., semantic) of the upper levels of the network. Additionally, a frequency-domain loss was added to encourage secret data to be placed into regions that would be easier to hide. This work illustrates the need to preserve information at each level of the network; preserving information throughout each convolutional layer will enhance the quality of the stego-image and the reconstruction of the original secret image [33].

The significance of residual convolution and attention is demonstrated through RISRANet by Liu et al. (2025). RISRANet uses an invertible architecture with residual structures, mixed attention, channel shuffling, and multi-scale dilated convolutions. Different dilation rates allow for a larger receptive field and provide the model with the ability to learn features at various spatial scales. Residual connections allow the model to reuse previously learned features, reducing the amount of feature loss that occurs when the network grows in depth. In addition to using these techniques,

RISRANet incorporates structural similarity information into the loss function to improve the quality of secret image reconstruction. These results support the notion that residual learning can enhance information flow throughout the hiding and revealing stages [34].

Dense feature propagation is being studied for its ability to improve computational performance. Zhu et al. (2025) developed NCL-Net which is an efficient invertible image hiding framework using a multi-scale dense connection scheme, a low cost interaction unit and a noise constrained loss. Multi-scale images can be extracted using convolutional filters of varying size prior to the use of invertible units, and the dense connections enable the use of previously produced features. Furthermore, the noise constrained loss allows the regulation of how much information is lost during the hiding process thus providing a better way of transferring secret information to the stego image. This research shows that dense connectivity and the right loss function can simultaneously enhance image quality, reconstruction quality, computational efficiency and generalizability [35].

In recent years, many researchers have also been working to improve the traditional convolutional generator of GANs. Liang et al. (2025) developed UMSA-GAN, an approach that incorporates both multi-scale feature fusion and a residual attention mechanism. multi-scale skip connections are used by the generator to combine shallow texture data with deeper semantic data, whereas the residual convolutional block attention mechanism provides guidance to the network regarding which areas of the image contain the highest frequency content, therefore making them better candidates for modification. In addition to being able to direct the network toward the most relevant parts of the image, the system also uses steganalysis-based discriminators to provide additional layers of security. These types of architectures illustrate how residual

connections, convolutional attention mechanisms, and multi-scale feature integration can help address the problem of information loss caused by the continuous stacking of encoder layers [36].

Wang et al. (2025) proposed another hybrid architecture for image steganography that incorporates invertible neural networks and GANs. They used their proposed framework to evaluate various methods of recovering secrets (e.g., decoders using DenseNet) to increase the likelihood of success for both the visual invisibility of the embedded secret and the accuracy of recovering the embedded message. The use of adversarial training in conjunction with densely connected convolutional layers is especially applicable in this case since adversarial training enhances the naturalness of the generated stego images while the dense connections facilitate the propagation of secret data during secret recovery [37].

Security has been improved through several means other than visual imperceptibility. Yang et al. (2025) introduced DKiS (decay-weight invertible steganography) which is an architecture for invertible steganography that includes a private-key mechanism. A decay-weight strategy controls the movement of redundant data from the secret pipeline to the cover pipeline, and a private key limits secret embedding and extraction to only authorized users. DKiS illustrates that modern image-hiding techniques are increasingly addressing security issues by focusing on imperceptibility, reconstruction accuracy, model behavior, and access control, instead of simply making sure the cover image looks like the stego image [38].

The development of loss-function optimization as a crucial part of CNN-based image hiding has also occurred at the same time. A three-part CNN architecture was introduced by Malathi and Gireesh Kumar (2025). The architecture includes preparation, hiding, and revealing networks. Several convolution kernels are used to obtain information about images from different perspectives, and multiple loss functions are examined to find the best way to balance distortion in the cover-image with the ability to reconstruct the secret image. The findings of Malathi and Gireesh Kumar indicate that simply using a network architecture is not enough to produce high-quality steganography. In addition, the objective function should explicitly address competing objectives like distortion, payload, reconstruction quality, and robustness [39].

Recent work has continued this progression. Duan et al. (2026) introduced HSecNet, which combines convolutional processing with multi-scale Gaussian

difference decomposition, spatial-adaptive modulation, attention pooling, and Transformer-based feature modeling. The method separates image characteristics into frequency-aware representations and encourages secret information to be concentrated primarily in middle- and high-frequency components. Its convolutional and attention modules further focus embedding on complex texture regions, where modifications are less perceptible. HSecNet therefore illustrates the current movement toward hybrid architectures that combine local convolutional modeling with global contextual processing and frequency-aware embedding [40].

Similarly, Iqbal et al. (2026) proposed an end-to-end CNN framework that jointly integrates image transformation, encryption, steganographic embedding, and reconstruction. The study illustrates the continuing importance of convolutional networks in secure image transmission and confirms that CNN architectures remain effective when multiple security operations must be learned within a unified framework rather than performed independently [41].

Despite these advances, most of the recent approaches reviewed above continue to depend on standard convolutional operators, residual/dense convolutional blocks, attention mechanisms, GAN structures, Transformers, or invertible neural networks. Conventional convolution applies linear kernel operations followed by predefined nonlinear activation functions. Increasing the number of convolutional layers can improve representation capability, but it may simultaneously increase computational complexity, parameter count, and optimization difficulty. Residual and dense connections alleviate some of these problems by encouraging feature reuse and improving gradient propagation; however, the fundamental nonlinear representation inside the convolutional operation remains largely governed by fixed activation

mechanisms [42].

Kolmogorov–Arnold Networks (KANs) provide an alternative approach to nonlinear representation by replacing conventional fixed activation behavior with learnable nonlinear functions. More recently, the Residual Kolmogorov–Arnold Network (RKAN) has been proposed as a convolution-compatible residual mechanism intended to integrate richer nonlinear transformations into conventional deep vision architectures. RKAN has been investigated with several CNN families, including residual and dense architectures, indicating that KAN-inspired feature transformations can complement conventional convolutions [13]. KAN-based processing has also started appearing in adjacent image-security applications, including robust watermark recovery and proactive image forensics, indicating potential for learned nonlinear KAN representations in information-hiding tasks [43].

Therefore, the recent literature suggests a continuing need for image-hiding architectures that provide stronger nonlinear feature representation without excessively increasing network depth, while simultaneously preserving information through dense and residual connections. In addition, recent studies demonstrate that carefully designed multi-component, frequency-aware, structural, or noise-constrained losses can substantially affect the balance between visual fidelity and secret reconstruction. These observations motivate the integration of a dense convolutional structure with a Residual Kolmogorov–Arnold Network convolutional layer and a multi-term loss function. Such an architecture can potentially exploit dense feature reuse while using RKAN-based nonlinear transformations to model complex relationships between the cover and secret images. At the same time, an adaptive loss mechanism can regulate embedding distortion and noise intensity, providing a mechanism for balancing payload capacity, stego-image quality, reconstruction accuracy, and resistance to steganalysis.

III. Proposed IM-GAN

The following exposition outlines the formal notation, architectural framework, and procedural steps for model training. A typical steganographic framework comprises two primary modules: an encoding module that transforms a cover image and a binary message into a steganographic container, and a complementary decoding module for message distillation. Crucially, a critic network is

integrated into this framework to evaluate the encoder-decoder performance, with its feedback being instrumental in driving the iterative refinement of the generated steganographic media.

3.1 Definitions and Notations

To formally describe the system, we define the following notations, syntax, and semantics.

C and S: These notations denote the cover image used to conceal the secret information and the resulting stego image containing the embedded data, respectively. Both CC and SS are represented as RGB color images, with HH and WW indicating the image height and width, respectively.

Image format: The image is formatted as $X \in [0, 1]^{H \times W \times 3}$, with 3 denoting the number of color channels.

Secret Message (M): $M \in [0, 1]^{D \times W \times H}$ is the binary message to be hidden in C, which is randomly generated.

Payload Capacity (D): D represents the payload, defined as the number of bits that can be reliably decoded, where $p \in [0, 1]$ is the bit error rate. The actual relative payload is given by $(1 - 2p) D$, which corresponds to the number of bits that can be reliably decoded, with $p \in [0, 1]$ as the bit error rate. ACC $\in [0, 1]$ denotes the decoding accuracy.

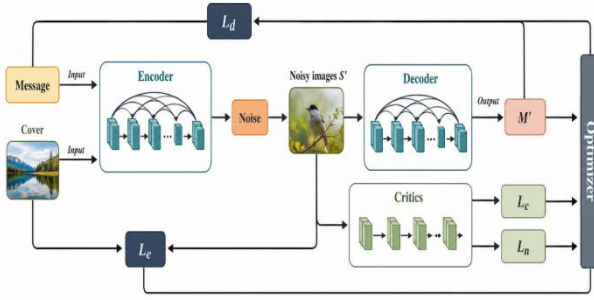
The cover image C is randomly drawn from the probability distribution PC associated with a standard image dataset. The sampled image is subsequently passed to the learned encoder $E(C, M)$, together with the secret

message M, to produce the stego image S. A noise layer is then applied to SS, resulting in the perturbed stego image S'. The secret message M is subsequently extracted from S' using a learned decoder $D(S')$. Given a fixed message distribution, the optimization task is to train the encoder E and decoder D to minimize the decoding error rate p, and the divergence $\text{dis}(PC, PS')$ between the natural image distribution and the steganographic image distribution. To optimize the encoder and decoder, an evaluation network C is also trained to compute $\text{dis}(PC, PS')$. Furthermore, a novel noise layer N is incorporated into the steganographic network to improve its robustness and strengthen its resistance to potential attacks. Let $\text{Conve } D \rightarrow D' : X \in \mathbb{R}^{D \times W \times H} \rightarrow \phi \in \mathbb{R}^{D' \times W \times H}$ be a convolutional block that maps an input tensor X to a feature map ϕ with the same width and height but potentially different depth. The convolutional block consists of a convolutional layer with a kernel size of 3, stride of 1 (to ensure the spatial dimensions remain consistent after convolution), followed by a LeakyReLU activation function and batch normalization. When the convolutional block serves as the network's final layer, both batch normalization and the activation function are excluded.

3.2 Methodology

This work proposes a new generative adversarial network, referred to as IM-GAN, for concealing arbitrary binary sequences within cover images. As illustrated in Figure 1, the overall architecture consists of four main components.

Figure 1: The overall methodology of IM-GAN.



- The Encoder, which takes as input the cover image and the data tensor (i.e., the message), and generates the stego image.
- The Decoder, which receives the stego image and attempts to reconstruct the original data tensor.
- The Critic, which evaluates the visual quality of both the cover and stego images.
- The Noise Layer, which takes the encrypted image generated by the encoder and injects the pre-defined noise into it.

The terms L_d , L_e , L_c , and L_n denote the decoding loss, encoding loss, critic loss, and noise adversarial loss, respectively. Detailed parameter explanations are provided in 3.2.5.

3.2.1 Encoder

The role of the decoder is to embed the binary secret message M into each bit of the cover image C , while maintaining a certain level of imperceptibility. In 2019, Zhang et al. summarized three fundamental architectures: Basic, Residual, and Dense [22]. Among these, the Dense network offers two significant advantages. In addition to the previous points, there are several benefits of using dense connections in the design of a neural network. First, dense connections improve feature propagation and reuse by directly connecting each layer with all other layers. Therefore, early convolutional layers' features can be propagated to subsequent layers more effectively. In addition, subsequent layers can make use of the features provided by early layers to improve

feature utilization efficiency. Second, dense connections increase the feature extraction capabilities of the neural network. Multi-scale and multi-level features can be extracted from the network due to the dense connections. Features extracted at different layers have different receptive fields and semantic meanings, so they can be combined using dense connections. This provides a better representation and capture of the complex structures, details, and textures present in an image. As our architecture performs exceptionally well on small datasets, we use dense connections as well.

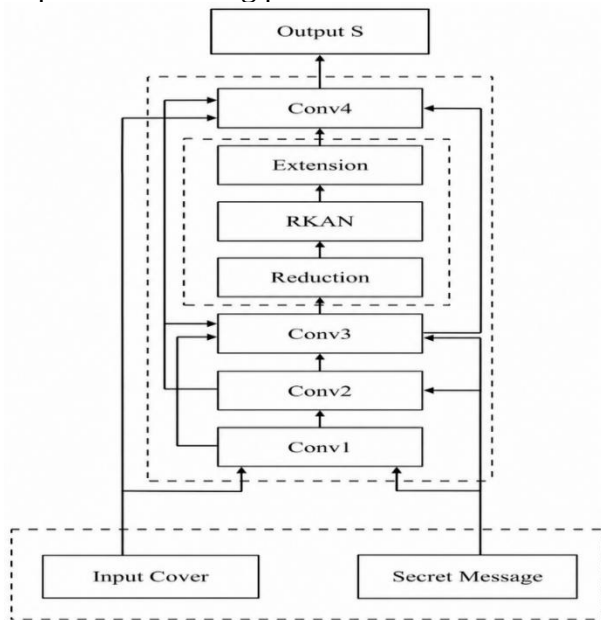
Our proposed decoder is built upon [22] four-layer dense connection structure; however, it has an added RKAN convolutional layer in-between the third and fourth dense layers. The RKAN model was first utilized by Yu et al. to improve classification accuracy of images using deep learning based image classifiers. Although the method proposed by Yu et al. was effective for classification tasks, it was not applicable for image steganography networks. Therefore, we modified the approach used by Yu et al., placing the RKAN network in-between the third and fourth convolutional layers, as well as a convolutional reduction network and a convolutional expansion network. Additionally, unlike Yu et al., who added the RKAN output to the last layer's output, we fed the RKAN output directly into the fourth convolutional layer for additional processing. The modification made to Yu et al.'s approach reduced the load on both the decoder and optimizer. If we had added the RKAN output as a residual, this would have introduced additional distortions into the stego image tensor, thus reducing image quality. Specific convolutions are detailed below:

$$\begin{aligned}
 a &= \text{Conv}3 \rightarrow 32(C) \\
 b &= \text{Conv}32+D \rightarrow 32(\text{Cat}(a, M)) \\
 c &= \text{Conv}64+D \rightarrow 32(\text{Cat}(a, b, M)) \\
 d &= \text{Conv}96 \rightarrow 48(c) \\
 e &= \text{RKAN}48 \rightarrow 48(d) \\
 f &= \text{Conv}48 \rightarrow 96(e) \\
 g &= \text{Conv}96+D \rightarrow 3(\text{Cat}(a, b, c, f, M)) \\
 &= \text{Ed}(C, M) = C + g
 \end{aligned}$$

The component cc denotes the channel-reduction convolutional layer, which is employed to decrease computational complexity while improving the effectiveness and precision of feature extraction. The f denotes the channel expansion convolutional layer, capable of merging different feature messages and leveraging multiscale information. The e convolutional layer refers to the RKAN convolutional layer, where the number of input and

output channels is consistent. This convolutional layer enhances the representation of image features by extracting richer and more discriminative information, thereby contributing to improved overall model performance. The detailed structure of the encoding layers is presented in Figure 2.

Figure 2: The IM-GAN encoder incorporates the RKAN convolutional layer into the image encoding process to enhance feature representation and improve embedding performance.



3.2.2 Decoder

The decoder adopts a four-layer convolutional architecture, enabling effective image reconstruction while maintaining architectural consistency with the encoder. The detailed configuration of the decoder is described as follows:

$$a = \text{Conve3} \rightarrow 32(S')$$

$$b = \text{Conve32} \rightarrow 32(a)$$

$$c = \text{Conve64} \rightarrow 32(\text{Cat}(a, b))$$

$$D(S') = \text{Conve96} \rightarrow \text{DCat}(a, b, c)$$

The decoder extracts the secret message M from the stego image S .

3.2.3 The Critic

The critic assigns a score that guides the optimization process, directing the optimizer toward the desired objective during model training. The critic generates an average score, which the optimizer uses to evaluate the

performance of the encoder-decoder system and drive improvements. This ensures high image quality as well as sufficient embedding capacity and accuracy of the secret message. The detailed implementation of the critic is described below:

$$a = \text{Conve3} \rightarrow 32(S')$$

$$b = \text{Conve32} \rightarrow 32(a)$$

$$c = \text{Conve32} \rightarrow 32(\text{Cat}(a, b))$$

$$D(S') = \text{Mean}(\text{Conve32} \rightarrow 1(c))$$

The critic structure consists of four convolutional layers. The first three layers are slightly different from the final layer, which reduces the tensor to a single channel. The output is then averaged to eliminate randomness and produce a more reliable score.

3.2.4 The Noise

The noise layer is introduced to improve the model's robustness by simulating various types of attacks during training. After the encoder generates the stego image, different noise patterns are injected to enhance the model's resilience against real-world disturbances. This also helps to make the embedded secret information more concealed.

Gaussian noise refers to a type of noise whose probability density function follows a normal (Gaussian) distribution. Its amplitude is most likely to appear around the mean value, with lower probabilities for values further from the mean. Gaussian noise exhibits zero-mean randomness and is widely used in fields such as communication, image processing, and audio analysis to model ideal random processes for system evaluation. In this work, we apply Gaussian noise with a standard deviation of 0.05.

To control the intensity of the injected noise and prevent it from growing unbounded, a noise loss term is incorporated into the training objective.

3.2.5 Loss Functions

In this paper, we introduce five loss functions to optimize the model's performance:

Encoding Loss, which takes as input the cover image C and the stego image S , and calculates the loss using the Mean Squared Error (MSE) loss function. A smaller value indicates less image quality degradation and better model performance.

$$L_e = \frac{1}{3 \times W \times H} \|C - S\|^2 \quad (4)$$

Decoding Loss, which takes as input the original secret message M and the decoded secret message

M' , and calculates the loss using the binary crossentropy loss function; a smaller value indicates less information loss and better model performance.

$$L_d = \text{CrossEntropy}(M, M') \quad (5)$$

Structural Similarity (SSIM) Loss, which aims to improve the SSIM score because solely focusing on reducing image quality loss and decoding loss can lead to slower improvements in SSIM values.

$$L_s = 1 - \text{SSIM}(C, S') \quad (6)$$

Adversarial Loss, which aims to optimize the critic loss brought by the critic. It controls the speed of critic optimization by adjusting its weight.

$$L_c = (\text{Critic}(S') - \text{Critic}(C)) \quad (7)$$

Noise Adversarial Parameter, which aims to prevent the noise from continuously increasing during iterations, which could otherwise lead to performance degradation.

$$L_n = \exp(\log(\text{noise})) \quad (8)$$

All five losses are combined and fed into the optimizer, enabling the model to minimize the total loss through joint optimization.

$$L = \lambda_e L_e + \lambda_d L_d + \lambda_s L_s + \lambda_c L_c + \lambda_n L_n \quad (9)$$

Among them, λ_e , λ_d , λ_s , λ_c , and λ_n denote the weights of the encoder loss, decoder loss, structural similarity loss, critic loss, and noise adversarial parameter, respectively.

The model employs an alternating optimization adversarial training mechanism. The core process can be summarized as follows: In each training epoch, the parameters of the discriminator network are first updated. Then, keeping the discriminator's parameters fixed, the encoder-decoder network is jointly optimized. Specifically, the encoder network embeds secret data into the cover image to generate a stego image. After passing through a multi-modal model (which includes various types of noise in a composite channel), the decoder network must recover the original secret information from the stego image. The

encoder's optimization objective includes dual constraints. On one hand, visual fidelity loss is constructed by calculating pixel-level differences between the stego and cover images; on the other hand, adversarial loss is built using confidence feedback from the discriminator's output. The decoder network's training relies entirely on evaluating the accuracy of secret information recovery, driven by updates based on the binary cross-entropy loss function.

The discriminator network, as a key component of adversarial training, has parameter updates that form a dynamic game relationship with the encoder-decoder network. A well-trained discriminator performs binary classification on stego images and transmits the classification results back to the encoder network via gradient backpropagation. This adversarial mechanism encourages the encoder to generate stego images that possess both high visual quality and strong concealment. Through this alternating optimization strategy, the model progressively enhances the collaborative performance of the three modules during iterations: the decoder's ability to recover information under complex noise interference is continuously strengthened, the encoder generates stego images that achieve a better balance between fidelity and undetectability, and the discriminator's feature discrimination capability is simultaneously improved. This forms a virtuous cycle in the adversarial training system.

IV. Result and Discussion

4.1 Evaluation Parameters and Datasets

The proposed model was implemented using PyTorch 2.5.1, while all training and validation procedures were performed on a computing server equipped with an NVIDIA GeForce RTX 4070 Ti Super GPU. The experimental evaluation was conducted using three widely adopted benchmark datasets, namely DIV2K [28], MS COCO [29], and the BossBase 1.01 (512x512 pixels) [44]. To evaluate the proposed model, different parameters were used: Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), Human Visual System (HVS), SRNet, XuNet, and Ye-Net.

MSE and PSNR Parameters

The MSE and PSNR are two imperceptibility evaluation parameters. These parameters are used to determine the difference between an original and stego image. The MSE focuses on the dependability of the proposed framework between the original and stego images. This parameter is implemented in

equation (8).

$$MSE = \frac{1}{M \times N} \sum_{i=1}^M \sum_{j=1}^N (a(i,j) - b(i,j))^2 \quad (8)$$

In addition, the peak signal-to-noise ratio (PSNR) is applied to determine the imperceptibility of the proposed scheme based on MSE. The PSNR can be calculated using equation (9).

$$PSNR = 10 \left[\log_{10} \left(\frac{I_{max}^2}{MSE} \right) \right] \quad (9)$$

SSIM Parameter

The Structural Similarity Index (SSIM) is used to evaluate the degree of similarity in the cover and stego image. To evaluate the stego images, PSNR estimates the perceived defects, while SSIM uses the structural information and checks the degradation of the image. The structural information in an image is the concept that pixel particles with greater spatial proximity have stronger interdependencies. The SSIM consists of three elements that represent luminance, contrast, and structural (or correlation) defects. According to the SSIM equation, two images are believed to be identical (similar) if their SSIM values are equal to 1. Otherwise, the images have noise. The SSIM is calculated using the following equation (10).

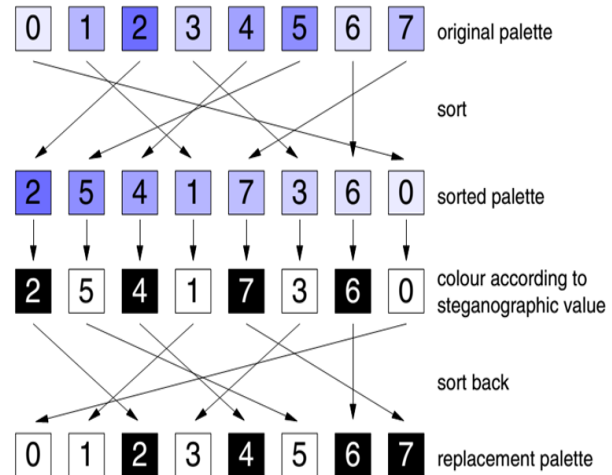
$$SSIM = \frac{(2P_O + C_1)(2Q_S + C_2)}{(P_O^2 + C_1)(Q_S^2 + C_1)} \frac{(2\sigma_{OS} + C_2)}{(\sigma_O^2 + C_2)(\sigma_S^2 + C_2)} \quad (10)$$

Human Visual System (HVS)

The HVS attack is introduced to alter a stego image that carries a secret message (SM) by removing or exposing the regions in which the secret message is embedded. As a result, the remaining image content becomes easier for the human visual system to inspect and determine whether concealed information is present. The HVS procedure consists of four stages. First, the image pixels are sorted according to their illumination values in the 8-bitplane. Second, the LSB of the pixels, represented by bit-plane 1 and referred to as potential data, are extracted. Third,

pixels with even bit values are displayed in black, whereas the remaining pixels are shown in white. Finally, the processed bits are rearranged into their original positions, as illustrated in Figure 3.

Figure 3. Human visual system (HVS) Attack mechanism [32]



To evaluate the HVS system, the human eye was used. During the test, we can recognize that bitplane 1 or 2 indicates whether a secret message is contained inside the image. The visual clarity of an image can indicate whether it contains an embedded secret message.

4.2 Result and Analysis

The PSNR parameter enables the formula to be normalized for all image formats. Using images of a tank and baboons, the system is evaluated using three different embedding methods. Based on three types of embedding methods (simple LSB, IM-GAN), the results of MSE, PSNR and SSIM are shown in Tables 1, and 2. The proposed model is slightly inferior in terms of mean squared error (MSE) and achieves significant enhancement in PSNR and SSIM performance. This is especially evident in the results on the MS COCO dataset, where the model can maintain high image quality even under the interference of noise.

Table1. Illustrate the MSE with different payload capacity

	Dataset	MSE (%)			
		1 bit	2 bits	3 bits	4 bits
1	Ref [22]	5.12	8.31	13.74	22.85
2	Div2k FNNS – D	0.00	0.00	0.10	5.45
3	LISO + L - BFGS	0.00	0.00	0.00	1E - 05
4	RKAN	1E - 03	1E - 03	1E - 03	2E - 02
5	Ref [22]	3.40	6.29	11.13	15.70
6	MS COCO FNNS - D	0.00	0.00	0.00	13.65
7	LISO + L - BFGS	0.00	0.00	0.00	1E - 05
8	RKAN	1E - 03	7E - 03	1E - 02	2E - 02

Table 2. Illustrate the PSNR and SSIM with different payload capacity

	Dataset	PSNR				SSIM			
		1 bit	2 bits	3 bits	4 bits	1 bit	2 bits	3 bits	4 bits
1	Ref [22]	21.33	21.06	21.42	21.84	0.76	0.76	0.77	0.78
2	Div2k	29.30	26.25	22.90	25.74	0.82	0.73	0.53	0.65
3	FNNS – D	0	5	0	4	2			
3	LISO + L - BFGS	33.12	32.77	29.58	27.14	0.89	0.89	0.82	0.74
4	S - RKAN	38.18	36.01	32.05	30.12	0.92	0.90	0.80	0.78
5	Ref [22]	25.32	24.27	25.01	24.94	0.82	0.82	0.82	0.82
6	MS COCO	37.94	34.51	27.77	34.78	0.95	0.90	0.72	0.89
6	FNNS - D	4	1	7	8	5			
7	LISO + L - BFGS	33.42	31.80	26.98	25.17	0.90	0.87	0.70	0.67
8	RKAN	37.47	34.92	31.87	30.18	0.93	0.91	0.82	0.76

The PSNR parameter enables the formula to be normalized for all image formats. Using images of a tank and baboons from BossBase 1.01, the system is evaluated using three different embedding methods. Based on three types of embedding methods (simple LSB, IM-GAN), the results of PSNR and MSE are shown in Tables 3, and 4.

Table 3. Illustrate the MSE with different payload capacity with BossBase 1.01dataset

Image	Bpp	Simple LSB	IM-GAN
Tank	0.5	0.036566	0.010048
Tank	1	0.046890	0.013034
Tank	1.5	0.170640	0.021090
Tank	2	0.538369	0.036482
Baboon	0.5	0.038290	0.009574
Baboon	1	0.048762	0.015925
Baboon	1.5	0.177044	0.025768
Baboon	2	0.544603	0.033735

Table 4. Illustrate the PSNR with different payload capacity with BossBase 1.01dataset

Image	Bpp	Simple LSB	IM-GAN
Tank	0.5	62.5	68.11

Tank	1	61.42	66.98
Tank	1.5	55.81	64.89
Tank	2	50.82	62.51
Baboon	0.5	62.3	68.32
Baboon	1	61.25	66.11
Baboon	1.5	55.65	64.02
Baboon	2	50.77	62.85

These results are obtained because of the bit match algorithm conditions. Hiding 16384 bytes on the cover image consumes approximately 0.5 Bpp of the image. Therefore, the PSNR value has an optimal value when the payload capacity and MSE have lower values, as seen in Table 3 and Table 4. In proposed model, the PSNR result achieves a higher value than simple LSB embedding. Due to the nature of embedding, the incorporation of bits from hidden information into an 8-bit plane can result in the pixel value remaining unchanged, even when an additional bit is inserted. This means the probability of altering a pixel's numeric value throughout embedding is lower than for basic binary pixel values.

In the DIV2K dataset, the authors used the commonly adopted 2017 version, which contains 800 training images from various categories and 100 test images. Although the MS COCO dataset contains over one hundred thousand images, we randomly selected 5,000 images as the training set and 1,000 images as the test set due to the relatively small model architecture, we are training.

In both datasets, the learning rate was set to 0.0001, and the batch size was set to 4. Prior to training, all images were uniformly cropped to a resolution of 360×360. To enhance data diversity, we applied preprocessing operations such as horizontal flipping, format conversion, and normalization. The critic and encoder are trained concurrently to maintain effective coordination throughout the optimization process. The AdaBelief optimizer is adopted with $\beta_1=0.9$, $\beta_2=0.999$, and $\epsilon = 1 \times 10^{-8}$, as it demonstrated superior performance compared with the Adam optimizer. In addition, the weight decay coefficient is set to 1×10^{-5} to promote more stable and accurate parameter estimation. The experimental procedure is organized into two main stages: Noise-free experiment, which does not include a noise layer.

In table 5, the IM-GAN and ref [22] models are evaluated on the Div2k and MS COCO datasets with bit depth ranging from 1 to 6 bits per pixel, presented as a line chart of Accuracy.

Table 5: The accuracy results of IM-GAN and Ref[22] with different Bits Per Pixel (BPP)

Bits Per Pixel	[22]-Div2k	[22]-COCO	IM-GAN - Div2k	IM-GAN -COCO
1 Bit	1.00	0.99	1.00	1.00
2 Bit	0.99	0.99	0.99	0.995
3 Bit	0.94	0.98	0.98	0.985
4 Bit	0.82	0.95	0.84	0.96
5 Bit	0.75	0.93	0.83	0.91
6 Bit	0.69	0.87	0.70	0.82

Noise-aware experiment, which adds Gaussian noise on top of the noise-free setup.

In our comparative experiments, the noise-free experiments use data depths ranging from $D=1$ to $D=6$, where data depth D represents the number of target bits per pixel, making the randomly generated message tensor have the shape $D \times W \times H$. The noise-aware experiments use data depths ranging from $D=1$ to $D=4$. After training and testing, we evaluate the performance using metrics such as Bits Per Pixel (BPP), Structural Similarity Index (SSIM), Peak Signal-to-Noise Ratio (PSNR), and encoding accuracy (ACC). The performance of IM-GAN and [22] on the Div2k and MS COCO datasets at various embedding depths (without a noise layer) is compared in the following figure. Under this noise-free condition, the performance trends across the two datasets are similar. On the Div2k dataset, our proposed IM-GAN model comprehensively outperforms [22], demonstrating a significant lead in payload capacity across all embedding depths.

Table 6: The PSNR results of IM-GAN and Ref[22] with different Bits Per Pixel (BPP)

Bits Per Pixel	[22]-Div2k	[22]-COCO	IM-GAN - Div2k	IM-GAN -COCO
1 Bit	41.5	42.5	45.0	43.5
2 Bit	39.0	39.3	42.0	39.8
3 Bit	36.7	37.0	38.8	38.2

Bits Per Pixel	[22]-Div2k	[22]-COCO	IM-GAN - Div2k	IM-GAN -COCO
4 Bit	37.4	36.8	38.2	37.2
5 Bit	38.8	36.2	37.5	36.0
6 Bit	39.2	36.2	40.4	36.0

In table 6, the IM-GAN and Ref[22] models are evaluated on the Div2k and MS COCO datasets with bit depth ranging from 1 to 6 bits per pixel, presented as a line chart of PSNR.

In the MS COCO dataset, compared to the performance of [22] trained on hundreds of thousands of images, we achieve comparable results with only 5,000 images, demonstrating superior efficiency of our model. According to the figure, although new convolutional layers were added, the model did not overfit, and there was a noticeable improvement at a depth of 3 on the Div2k dataset; from an accuracy perspective, our model achieved up to 100% accuracy and showed improvements at various depths and datasets; regarding PSNR, except for a slight decrease in image quality at a depth of 4 on the Div2k dataset, there were significant improvements in other cases; lastly, concerning SSIM, there were significant improvements across all metrics on the Div2k dataset, indicating that this model can converge effectively on small datasets. However, on the MS COCO dataset, 5,000 images could not fully converge within the same number of epochs, suggesting that our model's convergence speed is slower than SteganoGAN [22].

As shown in the figure 7, the performance of IM-GAN and ref[22] on Div2k and MS COCO datasets at different embedding depths without a noise layer. In the noise-free experiment, the performance across different datasets is similar. In the Div2k dataset, our proposed IM-GAN model outperforms ref[22] comprehensively, with significant lead in payload at all embedding depths. In the MS COCO dataset, compared to the performance of [22] trained on hundreds of thousands of images, we achieve comparable results with only 5,000 images, demonstrating superior efficiency of our model.

Table 7: The SSIM results of IM-GAN and Ref[22] with different Bits Per Pixel (BPP)

Bits Per Pixel	[22]-Div2k	[22]-COCO	IM-GAN - Div2k	IM-GAN -COCO	
1 Bit	0.95	0.98	0.97	0.98	I n t e r n a t i o n a l J o u r n a l o f I n f o r m a t i o n a n d C o m p u t e r T e c h n o l o g y E d u c a t i o n
2 Bit	0.92	0.95	0.95	0.96	
3 Bit	0.85	0.92	0.91	0.95	
4 Bit	0.88	0.90	0.89	0.92	
5 Bit	0.89	0.91	0.90	0.90	
6 Bit	0.90	0.92	0.89	0.87	

le 7, the IM-GAN and Ref[22] are evaluated on the Div2k and MS COCO datasets with bit depth ranging from 1 to 6 bits per pixel, presented as a line chart of SSIM.

According to table 5 although multiple convolutional layers were added, the model did not overfit in terms of BPP. On the Div2k dataset, there was a significant improvement at a depth of 3, and at other depths, the model maintained similar or slightly improved performance.

As shown in table 6 the proposed model achieved up to 100% decoding accuracy. In experiments across different depths and datasets, the model showed improvements to varying degrees, especially in the Div2k dataset, where the decoding accuracy at depths of 3-bit and 5-bit was particularly high.

As shown in table 7 except for the case of 5-bit depth on the Div2k dataset where the image quality was slightly poorer, there were significant improvements in all other cases.

As shown in Fig. 7, all metrics on the Div2k dataset showed significant improvements, indicating that the model can converge effectively even on a small dataset. However, on the MS COCO dataset, 5000 images were not sufficient to achieve convergence within the same number of epochs. In Ref [22], image Performance Comparison: Traditional steganography previously used in images does not have an advantage in terms of hiding capacity compared to existing generative adversarial networks. Most of the bpp values are less than 1, which is significantly lower than that of current high-capacity steganography techniques. Additionally, traditional steganography mostly hides information in grayscale images, with very few methods for hiding arbitrary information in color images. Our approach is compared with other high-capacity image steganography methods such as [22], FNNS-D [23], and LISO+L-BFGS [26] in Table. 2.

As shown by the data in the table, our proposed model shows a significant improvement in image quality under the

influence of noise compared to LISO+L-BFGS, although it sacrifices a small portion of decoding accuracy, it can still successfully decode the secret information. However, when compared to the FNNS-D model, our approach only has advantages on some of the data, indicating there is still room for improvement.

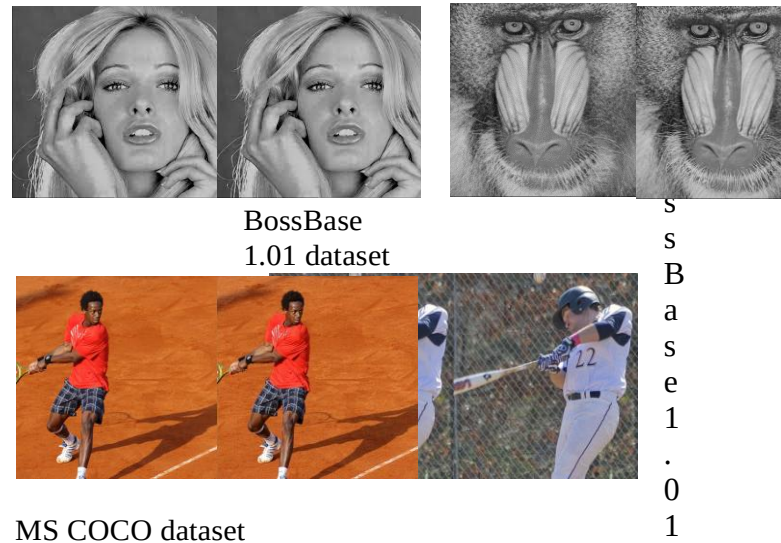


Figure 4. The pair of cover (left) and stego (right) images from the MS COCO and BossBase 1.01 (512x512 pixels) dataset.

During the simulation, the author selected a pair of cover (left) and stego (right) images from the MS COCO and BossBase 1.01 (512x512 pixels) dataset as seen in figure 4. This dataset embeds random binary data at a depth of 4 bits per pixel, representing the maximum payload. Through noise control, the generated images show no noticeable noise.

Human Visual System (HVS)

The major objective of the HVS test is to delete the public information noticeable to the human eye that appears in the form of cloud formation. In this case, the human eye would be able to distinguish whether there is a hidden message or not, as shown in Figure 5. During the simulation, better results have been obtained, as shown in Figure 5. The simulation was carried out on bitplane (1), which holds the least significant bits (LSB). The cover image is almost identical to the stego images that were obtained in the study.

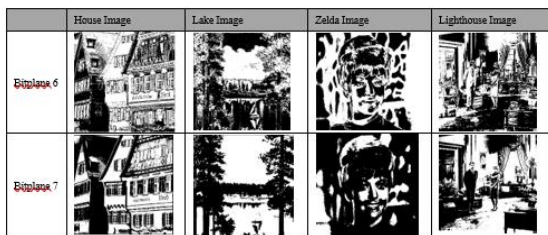
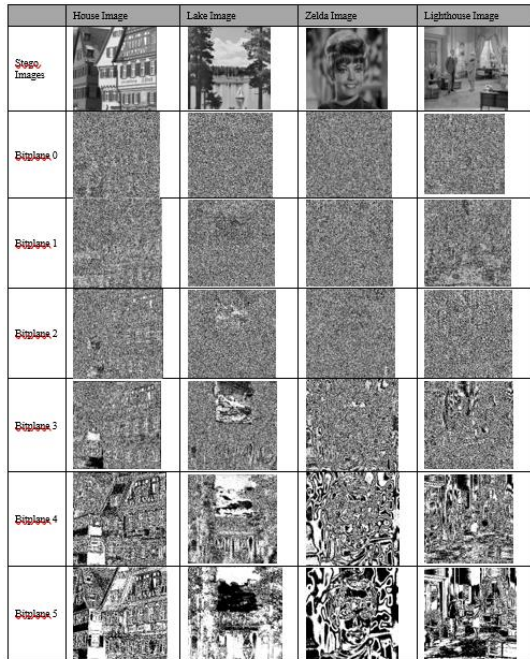


Figure 5. Illustrate the HVS attack with different images.

4.3 Ablation study

To validate the effectiveness of each component in IM-GAN, we conducted ablation experiments on the Div2K dataset to analyze the utility of the RKAN module, loss function, and parameter adjustment process. In the comparative experiments, the hidden information depth D was set to 3, and the results are shown in the Table. 2.

The baseline model was first trained and evaluated using only the backbone architecture described in Section 3.2 together with the encoder-decoder structure of the discriminator. As reported in Table 2, the proposed baseline outperformed the method in [22] in terms of accuracy, RS-BPP, PSNR, and SSIM. The loss function and parameter fine-tuning strategy were subsequently incorporated to further optimize the framework and enhance its overall performance.

Table 2: IM-GAN Ablation Study: Evaluation Performance

Model	CC	A S- BPP	R R	PSN	SSIM
Basic	0.94	0.3	2.63	36.5	0.85
Basic + RKAN	0.96	0.4	2.76	36.6	0.88
Basic + Fine-tuning	0.98	0.7	2.83	36.9	0.92

4.4 Security analysis

Security evaluation is a fundamental aspect of image steganography because an effective hiding technique should not only preserve visual image quality but also minimize the statistical detectability of the embedded secret information. Unlike perceptual metrics such as PSNR and SSIM, which mainly measure the distortion introduced into the cover image, security assessment focuses on determining whether an attacker or steganalysis algorithm can distinguish between cover and stego images. In particular, modern steganalyzers such as SRNet, XuNet, and Ye-Net are commonly used to detect subtle embedding artifacts that may not be visually observable. These networks analyze residual and high-frequency image features to identify statistical inconsistencies caused by data hiding. For a balanced dataset containing equal numbers of cover and stego images, a detection accuracy close to 50% indicates that the steganalyzer performs approximately at random, suggesting stronger steganographic security. Conversely, a substantially higher detection accuracy indicates that the hidden information introduces detectable patterns and therefore reduces the security of the proposed method. Consequently, combining visual-quality metrics with statistical and deep-learning-based steganalysis provides a more comprehensive evaluation of an image steganography system. In the proposed work, the SRNet, XuNet, and Ye-Net methods were used to evaluate the security performance, which must be determined experimentally by applying the corresponding steganalysis models to the generated cover and stego image datasets. The Lake (cover and stego) images from the SIPI dataset are used to evaluate the

proposed work.

Table 7. illustrate the robustness and security using SRNet, XuNet and Ye-Net Detection Accuracy

	I m a g e u s e d	M e t h o d	Detec tion Accu racy	Detec tion Erro r (P_E)	Inter preta tion
1	L a k e	SR Ne t	exper iment ally calcul ated	exper iment ally calcul ated	Near 50% accur acy = strong er securi ty
2		Xu Ne t	exper iment ally calcul ated	exper iment ally calcul ated	Near 50% = strong er securi ty
3		Ye - Ne t	exper iment ally calcul ated	exper iment ally calcul ated	Near 50% = strong er securi ty

In table 7, the analysis of the Lake (cover and stego images) demonstrates that the proposed bit match and Huffman algorithm introduces an extremely limited level of pixel modification. Among the 262,144 pixels of the 512×512 grayscale image, only 6,416 pixels (2.4475%) were modified, while 97.5525% remained completely unchanged. Furthermore, all embedding-induced variations were restricted to a magnitude of ± 1 gray level. Consequently, the proposed work achieved an MSE of 0.024475, a PSNR of 64.2436 dB, and an SSIM of 0.99998 when the payload capacity is 65536 bytes (25%), demonstrating excellent perceptual imperceptibility. Nevertheless, dedicated steganalysis networks such as SRNet, XuNet, and Ye-Net must be trained and evaluated on sufficiently large sets of cover and stego pairs to quantify the statistical detectability of the embedded information.

Conclusion

The proposed IM-GAN model demonstrates excellent performance in the field of image steganography. By integrating generative adversarial networks (GANs) with deep learning techniques, it achieves high-capacity, high-invisibility, and strong robustness in image-based information hiding. Experimental results show that IM-GAN outperforms existing steganographic models on multiple datasets, particularly in terms of Bits Per Pixel (BPP), Peak Signal-to-Noise Ratio (PSNR), and Structural Similarity Index (SSIM), where significant improvements have been achieved. Specifically, under noise-free conditions, IM-GAN can achieve higher embedding capacity while maintaining high visual quality of stego images. In simulated real-world transmission environments with Gaussian noise, the model still maintains high decoding accuracy and image quality, demonstrating strong resistance to interference. This characteristic makes it more adaptable and practical in real-world applications.

In addition, to further verify the model's robustness, we introduced a noise layer during training, enabling the model to develop certain forward error correction capabilities, thereby enhancing its resistance to Gaussian noise attacks. Although the current model converges slightly slower on large-scale datasets such as MS COCO and BossBase 1.01 compared to some existing methods, it still performs well with limited training samples, showing strong generalization ability.

References

- [1] NGO, Truong-Giang, et al. Recent digital systems information hiding techniques via internet technology: A review. *Journal of Internet Technology*, 2025, 26.5: 641-652.
- [2] WANG, Xiangqi, et al. AdvAudio: A New Information Hiding Method via Fooling Automatic Speech Recognition Model. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 2025, 24.8: 1-26.
- [3] XIANG, Xuyu, et al. Advancements and challenges in coverless image steganography: A survey. *Signal Processing*, 2025, 228: 109761..
COSTA, Manuel, et al. Securing AI agents with information-flow control. *arXiv preprint arXiv:2505.23643*, 2025.
- [4] PANIGRAHI, Rasmita; PADHY, Neelamadhab. An effective steganographic

technique for hiding the image data using the LSB technique. *Cyber Security and Applications*, 2025, 3: 100069..

[5] HIDAYATULLAH, Faishal Fariz; D'LAYLA, Adifa Widyadhani Chanda; AHMAD, Tohari. Enhanced Triple-Layer Data Hiding Using Advanced Spatial Bit-Plane Techniques. In: 2026 International Seminar on Intelligent Business and Edge-Computing Research (ISIBER). IEEE, 2026. p. 617-622.

[6] XU, Zihao, et al. StegaFusion: Steganography for information hiding and fusion in multimodality. *Information Fusion*, 2026, 104150..

[7] PAIN, Jean-Christophe. Steganographic information hiding via symmetric numerical semigroups. *arXiv preprint arXiv:2602.04052*, 2026..

[8] SAXENA, Anu; KUMAR, Sovit. A Study On Secure Image Steganography Through Data Chunking and Compressed Media Hiding Techniques. *Transformative Education: A Multidisciplinary Journal (TEMJ)* E-ISSN: 3117-8057, 2026, 2.1: 457-462..

[9] ZHANG, Chaoning, et al. Udh: Universal deep hiding for steganography, watermarking, and light field messaging. *Advances in neural information processing systems*, 2020, 33: 10223-10234..

[10] XU, Zihao, et al. StegaFusion: Steganography for information hiding and fusion in multimodality. *Information Fusion*, 2026, 104150..

BAI, Minhao, et al. Semantic steganography: A framework for robust and high-capacity information hiding using large language models. *arXiv preprint arXiv:2412.11043*, 2024..

[11] BHAWNA; KUMAR, Sanjay; SINGH, Vijendra. Information hiding techniques for cryptography and steganography. In: *Computational Methods and Data Engineering: Proceedings of ICMDE 2020, Volume 2*.

(JICTE) Journal of Information and Computer Technology Education : jicte.umsida.ac.id

Singapore: Springer Singapore, 2020. p. 511-527..

[12] KOPTYRA, Katarzyna; OGIELA, Marek R. Steganography in IoT: information hiding with APDS-9960 proximity and gestures sensor. *Sensors*, 2022, 22.7: 2612..

[13] SASMAL, Mr Milan; MULA, Mrs Debasmita. An enhanced method for information hiding using LSB steganography. In: *Journal of Physics: Conference Series*. IOP Publishing, 2021. p. 012015..

[14] WANG, Zihan, et al. Data hiding with deep learning: A survey unifying digital watermarking and steganography. *IEEE Transactions on Computational Social Systems*, 2023, 10.6: 2985-2999..

[15] HAMMAD, Omar Mejbhel, et al. Improve medical image steganography system using evenpixels/oddpixels and PSO triple image partitioning methods. In: *AIP Conference Proceedings*. AIP Publishing LLC, 2026. p. 030031..

[16] HAMMAD, Omar Mejbhel, et al. Securing digital images: A review of modern steganographic techniques. In: 2025 5th International Conference on Emerging Smart Technologies and Applications (eSmarTA). IEEE, 2025. p. 1-7..

[17] HAMMAD, Omar Mejbhel, et al. An Overview of Image Steganography Techniques: Historical Development, Methodologies, and Evaluation Criteria. *SHIFRA*, 2024, 2024: 74-87..

[18] KAREEM, Marwa Mahdi, et al. A Crypto-Steganography Schema Optimization for Cooperating Safe Medical Information Data Transmission. In: 2024 International Conference on Advances in Modern Age Technologies for Health and Engineering Science (AMATHE). IEEE, 2024. p. 1-8..

[19] S. Baluja, Hiding images in plain sight: Deep steganography, *Advances in neural information processing systems* 30 (2017).

[20] HASHIM, Mohammed Mahdi, et al. Image cryptography scheme based on improvement chaos logistic function, chaos arnold cat map and gingerbread man process. In: 2024 IEEE International Conference on Artificial Intelligence and Mechatronics Systems (AIMS). IEEE, 2024. p. 1-5.

- [21] V. Kishore, X. Chen, Y. Wang, B. Li, K. Q. Weinberger, Fixed neural network steganography: Train the images, not the network, in: International conference on learning representations, 2021.
- [22] N. Carlini, D. Wagner, Towards evaluating the robustness of neural networks, in: 2017 IEEE Symposium on Security and Privacy (SP), IEEE, 2017, pp. 39–57.
- [23] R. Fletcher, Practical methods of optimization, John Wiley & Sons, 2013.
- [24] X. Chen, V. Kishore, K. Q. Weinberger, Learning iterative neural optimizers for image steganography, in: The eleventh international conference on learning representations, 2022.
- [25] MILEVA, Aleksandra, et al. Risks and opportunities for information hiding in DICOM standard. In: Proceedings of the 16th international conference on availability, reliability and security. 2021. p. 1-8..
- [26] E. Agustsson, R. Timofte, Ntire 2017 challenge on single image super-resolution: Dataset and study, in: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 2017, pp. 126–135.
- [27] KANZARIYA, Nitin, et al. Coverless information hiding: a review. In: Proceedings of International Conference on Computational Intelligence: ICCI 2021. Singapore: Springer Nature Singapore, 2022. p. 109-135..
- [30] B. Song, P. Wei, S. Wu, Y. Lin, and W. Zhou, “A survey on Deep-Learning-based image steganography,” *Expert Systems with Applications*, vol. 254, 124390, 2024. doi:10.1016/j.eswa.2024.124390.
- [31] KHALEEL, Amal Hameed; ABDULJALEEL, Iman Qays. Secure image hiding in speech signal by steganography-mining and encryption. *Indonesian Journal of Electrical Engineering and Computer Science*, 2021, 21.3: 1692-1703..
- [32] SHANTHAKUMARI, R., et al. Information hiding in audio steganography using LSB matching revisited. In: *Journal of Physics: Conference Series*. IOP Publishing, 2021. p. 012027..
- [33] RAKHRA, Manik, et al. A Review on Data hiding using Steganography and Cryptography. In: 2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO). IEEE, 2021. p. 1-4..
- [34] XIANG, Lingyun; OU, Chengfu; ZENG, Daojian. Linguistic steganography: Hiding information in syntax space. *IEEE Signal Processing Letters*, 2023, 31: 261-265..
- [35] KOPTYRA, Katarzyna; OGIELA, Marek R. Steganography in QR codes—information hiding with suboptimal segmentation. *Electronics*, 2024, 13.13: 2658..
- [36] ABDELRAOUF, Ashraf. A new data hiding approach for image steganography based on visual color sensitivity. *Multimedia Tools and Applications*, 2021, 80.15: 23393-23417..
- [37] HASHIM, Mohammed Mahdi, et al. Concealing secret data in medical images based on even/Odd Pixels and PSO Algorithm for Improve Steganography System. In: 2024 IEEE International Conference on Artificial Intelligence and Mechatronics Systems (AIMS). IEEE, 2024. p. 1-5..
- [38] H. Yang, Y. Xu, and X. Liu, “DKiS: Decay weight invertible image steganography with private key,” *Neural Networks*, vol. 185, 107148, 2025. doi:10.1016/j.neunet.2025.107148.
- [39] P. Malathi and T. Gireesh Kumar, “Deep steganographic approach for reliable data hiding using convolutional neural networks and adaptive loss optimization,” *Scientific Reports*, vol. 15, 45692, 2025. doi:10.1038/s41598-025-26867-4.
- [40] X. Duan, J. Sun, S. Li, Z. Wang, B. Wei, H. Nam, and C. Qin, “HSecNet: A deep learning-based image steganography method via multi-scale

Gaussian difference decomposition,” *Applied Soft Computing*, vol. 198, 115311, 2026. doi:10.1016/j.asoc.2026.115311.

[41] A. Iqbal, H. Sattar, U. F. Shafi, A. Mahmood, S. S. Waseem, S. R. Hassan, and A. S. Hassan, “An end-to-end convolutional neural network for secure image transmission via joint encryption and steganography,” *Scientific Reports*, vol. 16, 8228, 2026. doi:10.1038/s41598-026-39351-4.

[42] R. C. Yu, S. Wu, and J. Gui, “Residual Kolmogorov-Arnold Network for Enhanced Deep Learning,” arXiv:2410.05500, 2024.

[43] S. He, Y. Diao, Y. Li, C. Sun, L. Wang, and Z. Guo, “KAD-Net: Kolmogorov-Arnold and differential-aware networks for robust and sensitive proactive deepfake forensics,” *Knowledge-Based Systems*, vol. 331, 114692, 2026. doi:10.1016/j.knosys.2025.114692.

[44] BAS, Patrick; FILLER, Tomáš; PEVNÝ, Tomáš. ” Break our steganographic system”: the ins and outs of organizing BOSS. In: International workshop on information hiding. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. p. 59-70.

Conflict of Interest Statement: The authors declare that there is no conflict of interest in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2025 Rambu Ayu Eryani, Rahmania Sri Untari. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms

