Advances in Science and Technology Research Journal

Advances in Science and Technology Research Journal, 2025, 19(8), 148–164
https://doi.org/10.12913/22998624/204419
ISSN 2299-8624, License CC-BY 4.0

Received: 2025.02.07
Accepted: 2025.05.27
Published: 2025.06.25

Reversible image steganography using transformer-based latent embedding

Olga Veselska[1], Ruslana Ziubina[1*]

[1] Department of Computer Science and Automatics, University of Bielsko-Biala, 43-306 Bielsko-Biala, Poland
* Corresponding author's e-mail: rziubina@ubb.edu.pl

ABSTRACT

Steganography, the practice of concealing information within media, has evolved significantly with advancements in deep learning. This paper presents a novel reversible image steganography framework based on transformer architectures. The proposed method embeds secret messages into the latent representation of an image obtained through a transformer encoder. The decoder, implemented as an inverse transformer network, enables the reconstruction of both the original image and the hidden message. This approach leverages the attention mechanism to enhance feature extraction, allowing for high embedding capacity while maintaining imperceptibility and robustness. Unlike traditional methods, it ensures full reversibility – a critical requirement in domains such as digital forensics and medical imaging. Experimental results demonstrate that the proposed system achieves high peak signal-to-noise ratio (PSNR) and message recovery accuracy, validating its effectiveness and practicality.

Keywords: image steganography; transformer architecture; latent representation; attention mechanism; deep learning; data hiding; image encryption, message recovery.

INTRODUCTION

The rapid digitization of global networks has transformed information into digital bits and bytes, making it an integral part of computer systems and communication channels. Critical information is now stored, processed, and transmitted digitally, exposing it to significant risks. Malicious actors target these systems to steal sensitive information or disrupt critical operations. To counteract these threats, cryptography, and steganography have emerged as pivotal tools in information security.

Modern cryptography offers a robust suite of techniques that ensure data confidentiality, integrity, and access control for legitimate users. Despite its strengths, cryptography faces challenges in real-world scenarios. Encrypted, authenticated, and digitally signed data can be cumbersome to access during critical decision-making moments. Furthermore, cryptographic methods cannot inherently ensure selective access control and encrypted messages can attract unwanted attention in environments where cryptographic practices are restricted [1].

Unlike cryptography, steganography addresses these limitations by concealing the very existence of secret messages. It embeds sensitive information into digital objects – such as text files, license keys, or media files – without arousing suspicion. Media files, including images, audio, and video, serve as particularly effective carriers due to their large data capacity and widespread usage. Advanced steganographic techniques can even embed messages so deeply that they remain intact after editing, resizing, printing, or scanning. These capabilities make image steganography a versatile and reliable approach for securing confidential information [2, 3].

With the integration of modern technologies, such as artificial intelligence, the field of steganography has witnessed significant advancements. In particular, deep-learning models like transformers, originally designed for natural language processing (NLP), have demonstrated exceptional performance in computer vision (CV) tasks [4].

This paper explores the potential of transformers for reversible image steganography, presenting a novel approach that leverages their ability to encode global dependencies and capture contextual relationships.

The reversible steganography method proposed in this paper embeds secret messages into media files while preserving their original quality, allowing for lossless recovery of both the hidden information and the original content. The integration of transformers into image steganography marks a step forward in achieving high-capacity, robust, and imperceptible data embedding. By addressing the limitations of traditional methods, this approach opens new avenues for secure communication and information protection in a variety of applications.

RELATED WORK

Traditional methods

Traditional steganography methods primarily utilize techniques such as least significant bit (LSB) embedding and frequency domain transformations. These methods aim to balance capacity, robustness, and imperceptibility, but often struggle to achieve all three simultaneously. The following is an analysis of the main approaches, their characteristics and limitations.

LSB-based methods embed secret data by altering the least significant bits of pixel values, typically within RGB or grayscale channels. These techniques are straightforward to implement and allow for embedding a large amount of data. However, they are highly sensitive to image manipulations such as compression, cropping, or noise, which can corrupt the embedded message. While LSB embedding may allow partial reversibility depending on the implementation, it is not inherently designed for full recovery. Additionally, slight pixel modifications can become noticeable in high-resolution or uniform-colored image regions. LSB techniques include methods such as simple LSB substitution, which directly modifies the least significant bits to embed data. Another approach is pseudorandomized LSB embedding, which enhances security by randomly selecting pixel positions for data embedding [5].

Frequency domain methods, such as those using the discrete cosine transform (DCT) and discrete wavelet transform (DWT), embed data into transformed coefficients of the image [6]. These methods are advantageous for their resistance to common image manipulations, including lossy compression (e.g., JPEG), as they modify less perceptible areas of the image. While frequency domain techniques improve robustness, they are not inherently reversible, as the original coefficients are typically altered irreversibly. Modifications are usually applied to frequency components that are less noticeable to the human visual system, such as high-frequency regions.

Generative adversarial networks (GANs) have emerged as powerful tools for steganography, enabling the embedding of large amounts of data without compromising perceptual quality. GANs excel in producing robust stego-images, as their adversarial training helps them adapt to perturbations or image manipulations. While GAN-based methods can achieve partial reversibility under specific conditions, full reversibility is not guaranteed by default. The generative capabilities of GANs ensure excellent visual quality, producing images that are nearly indistinguishable from natural ones. GAN-based methods utilize techniques such as modifying GAN-generated features to embed hidden messages. Additionally, adversarial training is employed to improve both robustness and imperceptibility, ensuring the stego-images remain indistinguishable from natural images [7]. The evaluation of traditional steganography methods is presented in Table 1.

Neural network-based steganography

Recent advancements in neural networks have led to the use of CNNs and RNNs for steganography. For example, hide and seek employs CNNs for robust image steganography. However, their temporal modeling capabilities in video remain limited. Deep neural networks (DNNs) offer remarkable capabilities in embedding information with high imperceptibility and robustness. However, many methods do not prioritize reversibility, which limits their applicability in sensitive domains such as medical imaging or digital watermarking.

CNN-based methods have become a cornerstone of modern steganography due to their ability to encode secret data into images with high imperceptibility. These approaches typically utilize encoder-decoder architectures, where the encoder embeds the secret data into a cover image, and the decoder retrieves it. By optimizing for

Table 1. Evaluation of traditional steganography methods

Method	Capacity	Robustness	Reversibility	Imperceptibility (PSNR, dB)
LSB-based techniques	Up to 12.5% of image size (e.g., 40 KB for 512 × 512 grayscale)	Low (bit errors ≈ 80% under JPEG compression Q ≤ 50)	Moderate (partial recovery)	35–40
Frequency domain (DCT, DWT)	1–5% of image size (e.g., 4–20 KB for 512 × 512 image)	High (robust to JPEG Q ≥ 30)	Low (irreversible)	40–50
GAN-based approaches	15–30% of image size (e.g., 1 MB for 512 × 512)	High (robust to cropping, resizing, noise)	Moderate (90–95% recovery)	45–55

imperceptibility, these models achieve robust and visually seamless data embedding [8].

GANs have been employed in steganography to enhance imperceptibility through an adversarial framework. In these methods, the generator embeds the secret message, while the discriminator evaluates the quality of the stego-image, pushing the generator toward producing highly realistic outputs. This dynamic ensures that the embedded information remains concealed from human perception while maintaining robustness to distortions [9].

Hybrid deep learning methods combine neural networks' strengths to achieve robustness and reversibility. These models often utilize separate networks for embedding and recovery, ensuring that the secret data and the original cover image can both be perfectly retrieved. Such methods are particularly relevant for sensitive applications, including medical imaging and digital watermarking, where reversibility is critical [10].

Attention mechanisms in steganography based architectures focus on embedding data in visually less sensitive regions of an image, guided by mechanisms such as saliency maps or attention layers. These methods are optimized for human perception, enabling high-capacity data hiding while maintaining minimal visual distortion. Additionally, they demonstrate robustness against manipulations like noise or compression. However, reversibility remains moderate, as these methods often rely on approximate reconstructions [11]. The evaluation of neural network-based steganography methods is presented in Table 2.

Machine learning methods, particularly those using neural networks, offer significant advantages in imperceptibility and robustness for steganography. However, many models sacrifice reversibility in favor of embedding capacity and robustness, limiting their use in applications like medical imaging.

Transformers in steganography

Transformers, originally developed for natural language processing, have gained traction in computer vision tasks due to their ability to capture global dependencies in data. Recent studies have explored their application in steganography, particularly for robust image encoding. However, the potential of transformers for reversible embedding, where both the original image and the hidden data can be perfectly recovered, remains underexplored. Below is an analysis of existing transformer-based methods in steganography.

Transformers, initially introduced by Vaswani et al. [12] for natural language processing tasks such as machine translation, have demonstrated exceptional performance due to their parallelization and ability to model long-range dependencies. This architecture quickly replaced LSTMs in NLP tasks and became the dominant model in the field. Recently, Transformers have also shown significant promise in computer vision (CV),

Table 2. Evaluation of neural network-based steganography methods

Method	Capacity	Imperceptibility (PSNR)	Robustness	Reversibility	Applications
CNN-based methods	Moderate (~5–10%)	High (50+ dB)	Moderate (resistant to noise, compression)	Low	Secure communication, DRM
GAN-based methods	High (~15–30%)	Very High (55+ dB)	High (robust to transformations)	Low	Covert communication, DRM
Reversible neural networks	Moderate (~5%)	Moderate (40–50 dB)	Moderate	High	Medical imaging, digital forensics
Attention mechanisms	High (~10–20%)	High (50+ dB)	High	Moderate	High-capacity data hiding

where their attention mechanisms enable efficient processing of high-dimensional visual data.

Dosovitskiy et al. [13] proposed the vision transformer (ViT) for image classification. ViT divides an image into 16 × 16 patches, treats each patch as a token, and processes them using a self-attention mechanism. The patches are flattened into one-dimensional vectors, allowing the model to learn global dependencies across the entire image. This innovation has been extended to other domains, including steganography.

In the context of steganography, transformers have been employed to encode secret data into images robustly. Their ability to extract meaningful latent representations makes them highly effective for imperceptible and resilient embedding. For example, ViTs have been used to identify the most relevant regions of an image for embedding hidden information while maintaining high robustness against distortions. However, most of these approaches lack reversibility, a limitation for applications requiring lossless recovery of both the secret data and the original image.

Tancik et al. [14] explored Vision transformers in robust watermarking systems, where they demonstrated excellent imperceptibility and resistance to manipulations such as noise and compression. Despite these strengths, these methods were not designed to recover the original cover image after the embedded data was extracted.

In [15], the author proposed a novel scheme to enhance steganography performance by leveraging Transformers' superior feature extraction capabilities. The method referred to as transformer–swim, employs a floating window mechanism that improves robustness and embedding efficiency. It was shown that this approach outperforms comparable state-of-the-art deep learning models, particularly in feature extraction for steganography tasks.

To address specific limitations, hybrid models combining transformers and convolutional neural networks (CNNs) have also been proposed. Wu and Liu [16] demonstrated a hybrid architecture where transformers process global features, and CNNs refine local embedding operations. This approach significantly improved robustness and imperceptibility but still did not fully achieve reversibility. The success of transformer-based models in steganography highlights their potential, but the lack of reversibility remains a challenge. Future research must focus on developing novel architectures or integrating reversible mechanisms, ensuring that both the hidden data and the original image can be perfectly recovered. The evaluation of transformer-based steganographic methods is presented in Table 3.

Transformers excel in global feature extraction and robustness. Their ability to embed large payloads while maintaining imperceptibility makes them promising for robust steganography applications, such as digital watermarking and covert communication. Most transformer-based methods focus solely on robustness and imperceptibility. Reversibility, a critical feature for sensitive applications like medical imaging or secure digital archiving, has not been fully addressed. This limits their broader adoption in applications where lossless recovery of the cover image is essential.

STEGO TRANSFORMER

Originally introduced for natural language processing tasks, transformers have revolutionized machine learning by offering unparalleled capabilities in modeling relationships within sequential and high-dimensional data. Their core functionality lies in the attention mechanism, enabling them to effectively capture local and global dependencies [17]. At the heart of the transformer is the self-attention mechanism, which computes the importance of each element in the input sequence relative to every other element. This mechanism ensures that the model focuses on relevant parts of the data while processing. In the context of images, transformers divide the image into patches (e.g., 16×16 blocks). Each patch is treated as a token, and its relationship with other

Table 3. Evaluation of transformer-based steganographic methods

Method	Capacity	Robustness	Reversibility	Imperceptibility (PSNR, dB)
Robust image encoding	High (~20–30%)	High (resistant to noise and compression)	Low (irreversible)	High (~50–55)
Vision transformers (ViT)	High (~25–35%)	High (robust to manipulations)	Low (irreversible)	High (~50–55)
Hybrid transformer models	High (~20–40%)	Very High (resistant to cropping, scaling)	Low to moderate (depends on hybrid design)	Very high (~55+)

patches is analyzed using self-attention. This allows the model to capture both local details and the global structure of the image.

Proposed in this work Stego transformer (StegoT) is a deep learning model that applies the Transformer architecture, originally developed for natural language processing (NLP), to steganography tasks [18].

To process the image a transformer encoder to capture spatial and contextual relationships between image patches. The image is divided into patches of size 16×16, and each patch is flattened into a vector. These flattened vectors are passed through a linear embedding layer to project them into a higher-dimensional latent space [20, 21]. Positional encodings are then added to each patch to retain the spatial structure of the image.

Let's break down Inputs to the attention mechanism using an example where the input consists of a matrix representation of an image and a secret message. These inputs are transformed into queries Q, keys K, and values V for the attention mechanism. The image is divided into patches 16×16 and each patch is flattened into a vector. These vectors are concatenated to form the image matrix:

$$Z_{image} \in R^{N \times d_{image}} \quad (1)$$

where: N – number of patches (64 for an 8×8 grid of patches), d_{image} – dimensionality of each patch (256).

Example:

$$Z_{image} = \begin{bmatrix} 1.2 & 0.5 & \dots & 0.9 \\ 0.7 & 1.1 & \dots & 0.4 \\ \vdots & \vdots & \ddots & \vdots \\ 0.3 & 0.8 & \dots & 1.5 \end{bmatrix}_{64 \times 256} \quad (2)$$

The secret message (e.g., a text string) is converted into a numerical tensor (e.g., ASCII or learned embeddings) and then aligned with the image representation:

$$Z_{message} \in R^{M \times d_{message}} \quad (3)$$

where: R – number of tokens in the message (16), d_{image} – dimensionality of each token (256).

Example:

$$Z_{message} = \begin{bmatrix} 0.2 & 1.1 & \dots & 0.6 \\ 0.9 & 0.4 & \dots & 0.7 \\ \vdots & \vdots & \ddots & \vdots \\ 0.5 & 0.3 & \dots & 1.0 \end{bmatrix}_{16 \times 256} \quad (4)$$

The two matrices are concatenated along the token axis to create a unified input:

$$Z = \begin{bmatrix} Z_{image} \\ Z_{message} \end{bmatrix} \in R^{(N+M) \times d} \quad (5)$$

Example:

$$Z = \begin{bmatrix} Z_{image} \\ Z_{message} \end{bmatrix} = \begin{bmatrix} 1.2 & 0.5 & \dots & 0.9 \\ \vdots & \vdots & \ddots & \vdots \\ 0.3 & 0.8 & \dots & 1.5 \\ 0.2 & 1.1 & \dots & 0.6 \\ \vdots & \vdots & \ddots & \vdots \\ 0.5 & 0.3 & \dots & 1.0 \end{bmatrix}_{80 \times 256} \quad (6)$$

Attention in the encoder allows the model to capture both local and global dependencies between image fragments using the multi-headed self-attention (MHSA) mechanism:

$$Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (7)$$

The combined matrix Z is linearly transformed into queries (Q), keys (K), and values (V) using learnable weight matrices W_Q, W_K, W_V:

$$Q = ZW_Q, \quad K = ZW_Q, \quad V = ZW_V$$

where: W_Q, W_K, $W_V \in R_d \times d_k$ and d_k is the dimensionality of the attention space.

For each input vector z_i (row of Z), we compute:

$$q_i = z_i W_Q, \quad k_i = z_i W_K, \quad v_i = z_i W_V$$

Example:

$$W_Q = \begin{bmatrix} 0.2 & 0.5 & \dots & 0.1 \\ \vdots & \vdots & \ddots & \vdots \\ 0.3 & 0.8 & \dots & 0.4 \end{bmatrix}_{256 \times 64} \quad (8)$$

Then:

$$Q = \begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_{80} \end{bmatrix}_{80 \times 64} \quad (9)$$

Similarly, K and V are computed. The multi-head version (H_{heads}) combines multiple perspectives of attention.

$$MHSA(Z) = Concat(head_1, head_2, \dots, head_H)W_o \quad (10)$$

Positional Encodings are added to tokens to retain spatial information within the flattened representation. This prepares the inputs for the scaled dot-product attention step. The attention mechanism computes weights for each token based on the similarity between queries (Q) and keys (K). The description of the algorithm is given below.

Step 1. Compute attention scores

The attention scores are computed as the dot product of Q and K^T, scaled by the square root of the dimensionality d_k to stabilize gradients:

$$Scores = \frac{QK^T}{\sqrt{d_k}} \quad (11)$$

where:

- $Q \in R^{(N+M)\times d_k}$
- $K^T \in R^{d_k\times(N+M)}$
- $Scores \in R^{(N+M)\times(N+M)}$

Let Q and K have small values for simplicity:

$$Q = \begin{bmatrix} 0.2 & 0.5 \\ 0.3 & 0.8 \\ \vdots & \vdots \end{bmatrix}_{80\times64}, K^T = \begin{bmatrix} 0.1 & 0.4 & \ldots \\ 0.6 & 0.3 & \ldots \end{bmatrix}_{64\times80} \quad (12)$$

The resulting scores:

$$Scores = \begin{bmatrix} 0.38 & 0.42 & \ldots \\ 0.56 & 0.48 & \ldots \\ \vdots & \vdots & \ddots \end{bmatrix}_{80\times80} \quad (13)$$

Step 2. Apply Softmax

To normalize the scores into probabilities, apply the softmax function row-wise:

$$AttentionWeights(i,j) = \frac{exp(Scores_{ij})}{\sum_{k=1}^{(N+M)} exp(Scores_{ik})} \quad (14)$$

Example for one row of scores:
Row of scores: [0.38, 0.42, ...] → Attention weights: [0.25, 0.27, ...]
Resulting matrix:

$$AttentionWeights = \begin{bmatrix} 0.25 & 0.27 & \vdots \\ 0.30 & 0.35 & \vdots \\ \vdots & \vdots & \ddots \end{bmatrix}_{80\times80} \quad (15)$$

Step 3. Compute weighted values

Multiply the attention weights with the values matrix V to produce the output representation:

$$Output = AttentionWeights \times V \quad (16)$$

where:

- $AttentionWeights \in R^{(N+M)\times(N+M)}$
- $V \in R^{(N+M)\times d_k}$
- $Output \in R^{(N+M)\times d}$

Let V be:

$$V = \begin{bmatrix} 0.3 & 0.7 \\ 0.6 & 0.2 \\ \vdots & \vdots \end{bmatrix}_{80\times64} \quad (17)$$

If a row of is [0.25, 0.27, ...]:

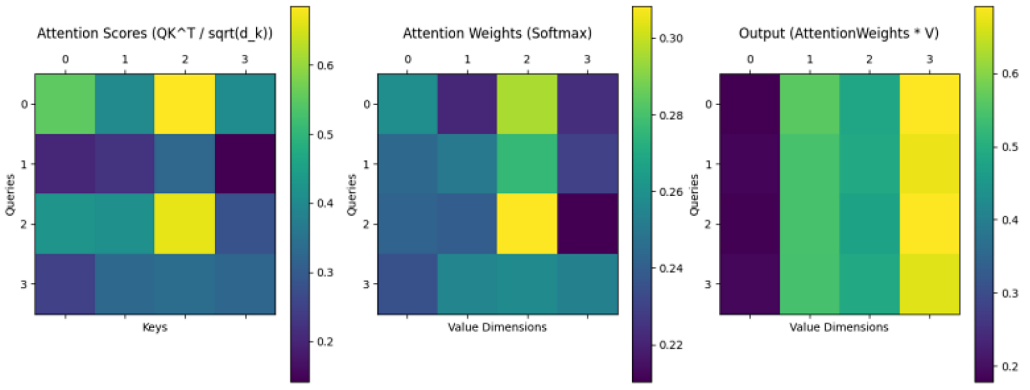
$$OutputRow = [0.25, \quad 0{,}27, \quad \ldots] \times \times \begin{bmatrix} 0.3 & 0.7 \\ 0.6 & 0.2 \\ \vdots & \vdots \end{bmatrix} = [0.42 \quad 0.38] \quad (18)$$

The output of the scaled dot-product attention is a matrix representing the weighted combination of image and message features:

$$OutputAttention = \begin{bmatrix} o_1 \\ o_2 \\ \vdots \\ o_{80} \end{bmatrix}_{80\times64} \quad (19)$$

This output will be passed through subsequent layers (e.g., feedforward or transformer decoders) for further processing. A visualization of the above described attention mechanism is presented in Figure 1 and shows:

- *Attention score matrix* $\left(\frac{QK^T}{\sqrt{d_k}}\right)$ – the matrix, which contains the raw alignment scores, is calculated by multiplying the query matrix by the transpose of the key matrix and scaling by the square root of the feature dimension (d_k).
- *AttentionWeights*(*Softmax*) – for evaluation are normalized for each query row using the



a) b) c)

Figure 1. The visualization of attention mechanism: a – attention score matrix, b – attention weights (Softmax), c – attention weights

softmax function to obtain a distribution over the keys**.**

- *Output*(*AttentionWeights*×*V*) – shows the attention weights computed by the weighted sum of the value vectors, which results in the final output of the attention mechanism.

The attention mechanism plays a pivotal role in the Stego Transformer framework, enabling effective embedding and extraction of secret messages within the latent space of images. The attention mechanism, whose block diagram is shown in Figure 2, a core component of Transformer architectures, dynamically focuses on the most relevant parts of the input data by computing pairwise relationships between all input elements. This capability is leveraged in the Stego Transformer to encode and decode secret messages with high fidelity and imperceptibility. In the encoding phase, the input image is divided into fixed-size patches, which are linearly projected into a high-dimensional feature space and augmented with positional encodings. The resulting embeddings are passed through multiple layers of self-attention, allowing the model to capture complex dependencies between patches. This step ensures that the latent representation incorporates both spatial and contextual information, which is crucial for embedding the secret message seamlessly without disrupting the perceptual quality of the cover image. The attention mechanism operates by computing query, key, and value vectors for each patch, enabling the model to identify and prioritize critical regions for embedding. By attending to relevant features, the Stego Transformer minimizes interference with the image's visual content, ensuring imperceptibility. Moreover, the attention mechanism helps distribute the embedded message across the latent space, enhancing robustness against distortions such as compression or noise. During the decoding phase, the transformer decoder employs cross-attention layers to reconstruct the secret message from the stego image's latent representation. The cross-attention mechanism aligns the embedded features with the original message's structure, enabling accurate recovery. This design ensures reversibility by maintaining the integrity of both the cover image and the secret message.

Proposed reversible steganography method

The proposed hiding network for Reversible Steganography Method is designed to embed secret messages into images while ensuring that both the secret message and the original image can be perfectly recovered. Originally intro duced for natural language processing tasks, transformers have revolutionized machine learning by offering unparalleled capabilities in modeling relationships within sequential and high-dimensional data. Their core functionality lies in the attention mechanism, which enables them to effectively capture both local and global dependencies. At the heart of the transformer is the self-attention mechanism, which computes the importance of each element in the input sequence relative to every other element [19-21]. This mechanism ensures that the model focuses on relevant parts of the data while processing. In the context of images, transformers divide the image into patches (e.g., 16×16 blocks). Each patch is treated as a token, and its relationship with other patches is analyzed using self-attention. This allows the model to capture both local details and the global structure of the image. Figure 3 illustrates the latent representation of an image under different processing techniques within the proposed Stego-Transformer framework. Subfigure (a) shows the original, unmodified input image, which serves as the reference for subsequent transformations.

Subfigure (b) presents a heatmap-based visualization of the image's latent representation, highlighting the spatial regions that encode stronger or more relevant semantic features after passing through the transformer layers. This

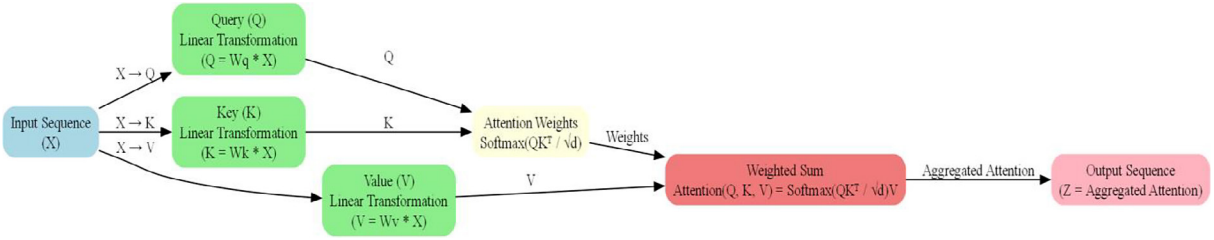


Figure 2. Flowchart of the attention mechanism

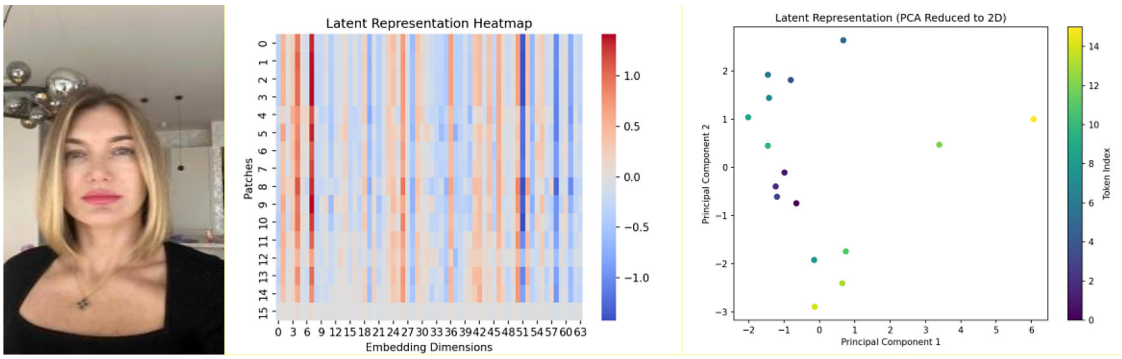


Figure 3. The latent representation of an image: a – original image, b – Latent reprezentation in heatmaps techniques, c – Latent reprezentation in PCA techniques

visualization reflects the internal attention distribution and the model's focus during the embedding process.

Subfigure (c) displays the principal component analysis (PCA) projection of the latent space, offering a reduced-dimensional view of the embedded features. This representation enables analysis of how information is compressed, organized, and separated within the latent space, providing insight into the structural properties of the encoded content.

This model ensures the original image can be perfectly reconstructed while securely retrieving the hidden message. The architecture leverages attention mechanisms to selectively embed information into regions of the image's latent space, optimizing imperceptibility, robustness, and reversibility. The StegoTransformer integrates attention mechanisms throughout its architecture to embed secret messages in an image's latent representation

The latent representation of the image produced by the transformer encoder operates in a high-dimensional space. Neural networks can learn an optimized representation of the message, capturing meaningful features while minimizing redundancy.

The architecture of the hiding network for reversible steganography with transformers

The proposed hiding network for reversible steganography with transformers is designed to embed secret messages into images while ensuring that both the secret message and the original image can be perfectly recovered. The hiding network in stegotransformer is designed to embed secret messages into the latent representation of an image, ensuring high imperceptibility, robustness, and reversibility. This approach introduces a novel reversible steganographic method that leverages transformer networks, where messages are encoded directly into the latent representation of the image. The latent representation, defined as a high-dimensional, abstract encoding of the image's structural and semantic features, is obtained by dividing the input image into non-overlapping patches, projecting these patches into a lower-dimensional latent space, and processing them through a multi-layer transformer encoder. This transformation ensures that the essential information from the original image is captured in a compact and manipulable form, enabling efficient and secure integration of the secret message while maintaining the image's integrity and quality. A latent image representation after processing by a transformer encoder is not a traditional image; instead, it is a high-dimensional numerical matrix or tensor. This latent representation captures the essential features of the image, such as spatial and contextual information, in a way that is meaningful for downstream tasks (e.g., embedding, classification, or steganography).

The proposed reversible transform steganography method (RSTM) architecture includes the following stages: input layer, feature extraction, message embedding, refinement and recovery, and output layer and is shown in Figure 4. The design ensures that the original image and the embedded secret message can be perfectly reconstructed by utilizing a multi-headed transformer self-capture mechanism for feature extraction and integration.

At the beginning, the original image and the secret message are fed to the input layer, where the data is initially processed into the form of matrices $C \in R^{H\times W\times 3}$, where H and W are the image dimensions and 3 denotes the RGB channels.

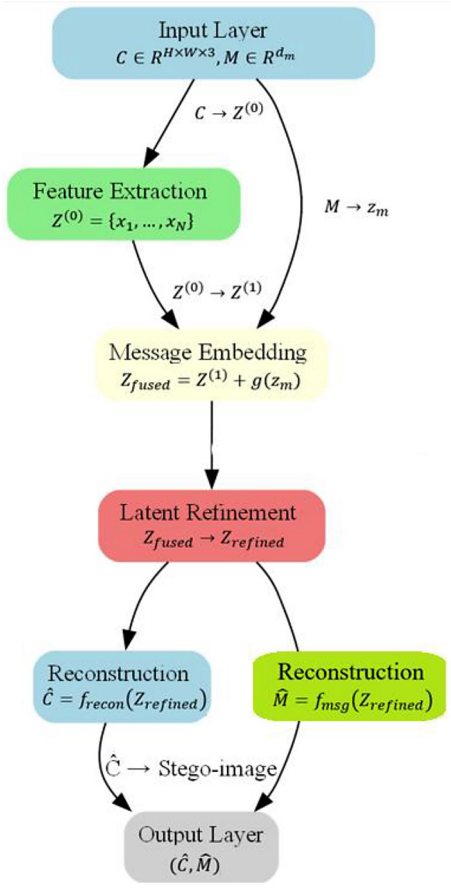


Figure 4. The architecture of the hiding network for reversible steganography with transformers

A secret message $M \in R^{d_m}$, represented as a binary or numeric sequence, is encoded using an embedding network:

$$z_m = f_{msg}(M), \tag{20}$$

where: f_{msg} maps M to a high-dimensional representation $z_m \in R^{d_m}$, compatible with the latent space of C.

The feature extraction stage, shown in Figure 4, uses the transformer encoder to process tokenized image slices. The multi-headed self-dual layers in Transformer are responsible for capturing global dependencies, which enables robust feature extraction from both spatial and spectral regions of the image.

For feature extraction, the cover image is partitioned into $N = \frac{H\times W}{P^2}$ non-overlapping patches of size $P\times P$, which are flattened into tokens:

$$x_i = Flatten(C[i])W_e, i = 1, \ldots, N \tag{21}$$

where: $W_e \in R^{(P^2\times 3)\times d_z}$ is a learnable projection matrix.

A Transformer encoder processes the tokens $Z(0) = \{x_1, x_2, \ldots, x_N$. The multi-head self-attention mechanism extracts both local and global features:

$$Attention(Q,K,V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{22}$$

where: $Q = ZW_Q$, $K = ZW_Q$, $V = ZW_V$ and W_Q, W_K, W_V are learnable projection matrices.

Positional encodings are added to the token embeddings to retain spatial information:

$$Z^{(1)} = Z^{(0)} + PE. \tag{23}$$

The secret message embedding z_m is integrated into the image's latent representation $Z^{(1)}$ using a fusion mechanism:

$$Z_{fused} = Z^{(1)} + g(z_m), \tag{24}$$

where: $g(\cdot)$ maps z_m to the dimensionality of $Z^{(1)}$.

This can be implemented as additive fusion $g(z_m)$ adds z_m to selected tokens, concatenative fusion $g(z_m)$ appends $g(z_m)$ as new tokens. The latent features with the embedded message undergo refinement using a Transformer decoder. The decoder refines the combined latent representation to prepare it for reconstruction Z_{fuzed}:

$$z^{(l+1)} = MHSA\left(LN(z^{(l+1)})\right) + z^{(l)}, \tag{25}$$

$$z^{(l+1)} = CrossAttention\left(LN\left(z^{(l+1)}\right), Z^{(1)}\right) + z^{(l+1)}, \tag{26}$$

$$z^{(l+1)} = FFN\left(LN\left(z^{(l+1)}\right)\right) + z^{(l+1)}, \tag{27}$$

where: $LN(\ )$ is layer normalization, FFN is a feedforward network, and l denotes the layer index.

During the reconstruction step, the refined latent $Z_{refined}$ representation is decoded to reconstruct the stego-image of $\hat{C}$:

$$\hat{C} = f_{recon}\left(Z_{refined}\right), \tag{28}$$

where: f_{recon} maps latent tokens back to the pixel space. To ensure imperceptibility, the reconstruction is optimized using the following losses:

Reconstruction loss

$$L_{recon} = \left\|C - \hat{C}\right\|_2^2, \tag{29}$$

ensuring minimal perceptual differences between C and $\hat{C}$.

Message recovery loss

$$L_{msg} = \left\|M - \hat{M}\right\|_2^2, \tag{30}$$

where: $\hat{M} = f_{msg}(Z_{refined})$ represents the extracted message.

Produces the stego-image, which visually replicates the original cover image with the hidden message embedded. Provides an inverse process to ensure lossless recovery of both the original image and the secret message during extraction.

The output layer at the output of $\hat{C}$ produces a stego-image, visually indistinguishable from the cover image C, with a message embedded in its hidden representation. The model guarantees that both the original image and the secret message can be recovered:

$$f_{recover}(Z_{refined}) = (\hat{C}, \hat{M}). \quad (31)$$

The Stego transformer's hiding network uses advanced transformer-based feature extraction, multi-head self-attention, and fusion mechanisms to embed messages imperceptibly. The architecture's design balances imperceptibility, robustness, and reversibility, making it suitable for secure communication applications. The total loss function is defined as:

$$L_{total} = \alpha L_{recon} + \beta L_{msg}, \quad (32)$$

where: α and β control the balance between image quality and message recovery.

The architecture of the extracting network for reversible steganography with transformers

The extraction network aims to recover the original image and embedded message from the encoded image. The architecture utilizes a decoder-transformer, ensuring reversibility and maintaining high fidelity of both the recovered image and the extracted message, is shown in Figure 5.

The encoded image is first divided into non-overlapping patches of size $P \times P$, resulting in patches. Each patch is flattened into tokens for processing by the transformer decoder:

$$\hat{x}_i = Flatten(\hat{C}[p_{i,1}:p_{i,2}, p_{i,1}:p_{i,2}, :]), \quad (33)$$
$$i = 1, 2, ..., N$$

The transformer decoder extracts features from the encoded image patches and fuses them with positional embeddings to recover spatial information:

$$z^0 = [\hat{x}_1 + p_1, \hat{x}_2 + p_2, \ldots, \hat{x}_N + p_N],$$

where: p_i are learnable positional embeddings.

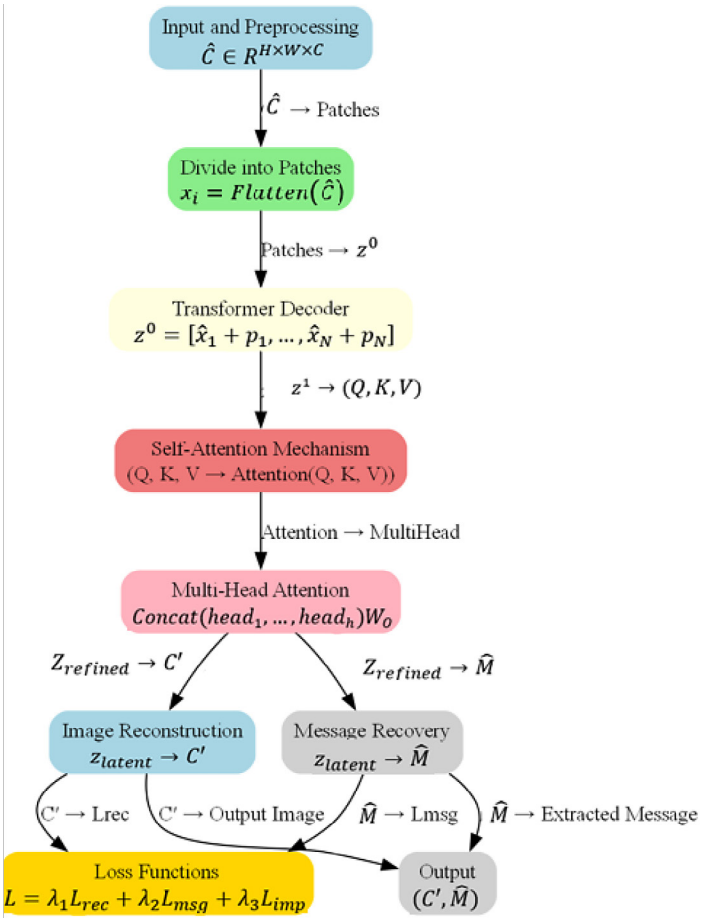


Figure 5. The architecture of the extracting network for reversible steganography with transformers

The self-attention mechanism in the decoder computes queries (Q), keys (K), and values (V) for each attention head: $Q = z^l W_Q$, $K = z^l W_K$, $V = z^l W_V$, where W_Q, W_K, W_V are learnable weights. The attention scores are then calculated as:

$$Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \quad (34)$$

where: d_k is the dimensionality of the key vector.

The multi-head attention aggregates information across multiple subspaces:

$$MSA(z^l) = Concat(head_1, \ldots, head_h)W_O, \quad (35)$$

where: W_0 is a learnable projection matrix, and h is the number of attention heads.

Finally, the features are refined using feed-forward layers and residual connections:

$$z^{l+1} = FFN\left(LauerNorm\left(MSA(z^l)\right)\right) + z^l. \quad (36)$$

Message and image reconstruction. The transformer decoder outputs latent features that are reshaped and passed through a refinement network to reconstruct the original image :

$$\hat{C} \rightarrow z_{latent} \rightarrow C'. \quad (37)$$

The reconstruction process minimizes the following loss to ensure fidelity:

$$L_{rec} = \|C' - C\|_2^2, \quad (38)$$

where: C' is the reconstructed image.

The latent features are simultaneously used to recover the embedded message M:

$$\widehat{M} = MLP(z_{latent}), \quad (39)$$

where: $\widehat{M}$ is the extracted message, and MLP is a multi-layer perceptron that maps latent features back to the message space.

The message recovery loss ensures the accurate retrieval of the original message:

$$L_{msg} = \left\|\widehat{M} - M\right\|_2^2. \quad (40)$$

The total loss function balances the reconstruction of the image and the recovery of the message, while ensuring imperceptibility of the steganographic process:

$$L = \lambda_1 L_{rec} + \lambda_2 L_{msg} + \lambda_3 L_{imp}, \quad (41)$$

where: L_{rec} – image reconstruction loss, L_{msg} – message recovery loss, L_{imp} – imperceptibility loss, which is defined as:

$$L_{imp} = 1 - SSIM(C', C), \quad (42)$$

and λ_1, λ_2, λ_3 are weights controlling the contribution of each loss component.

The final outputs of the network are the recovered original image C', which closely resembles C and the extracted secret message $\widehat{M}$, which perfectly matches M. This transformer-based extracting network ensures reversible steganography, enabling accurate recovery of both the cover image and the embedded message.

EXPERIMENT

To validate the proposed reversible image steganography framework, a structured experimental setup was developed using the CIFAR-10 dataset [23]. In this study, we specifically focused on the 'people' class, which comprises human-like figures often depicted in various poses and environments. This class was selected due to its rich structural diversity, including facial textures, clothing patterns, and complex backgrounds –factors that are particularly challenging for steganographic embedding. One of the reasons for selecting this class is the potential future application of this method for hiding confidential information directly within photographs of people. Such variability allowed us to assess how well the transformer-based encoder handles semantically rich and perceptually sensitive regions. The diversity in skin tones, edge patterns, and background clutter served as a useful benchmark for both imperceptibility and robustness analysis.

The embedded latent representation was processed by the transformer decoder to reconstruct the original image and recover the embedded message. Parameters such as peak signal-to-noise ratio (PSNR) were used to evaluate the quality of the reconstructed image, while the message recovery accuracy was validated through bit-wise comparison between the input and extracted messages.

The proposed reversible steganography method using transformers can be reduced to two main steps: embedding and extraction, as shown in Figure 6. In the embedding stage, the input data consists of a hidden image and a secret message . The hidden image is divided into non-overlapping regions and its features are extracted and these regions are encoded into hidden representations. At the same time, the secret message is processed and embedded into the latent features using the message embedding stage. Then, in the latent features refinement phase, they are combined and refined to

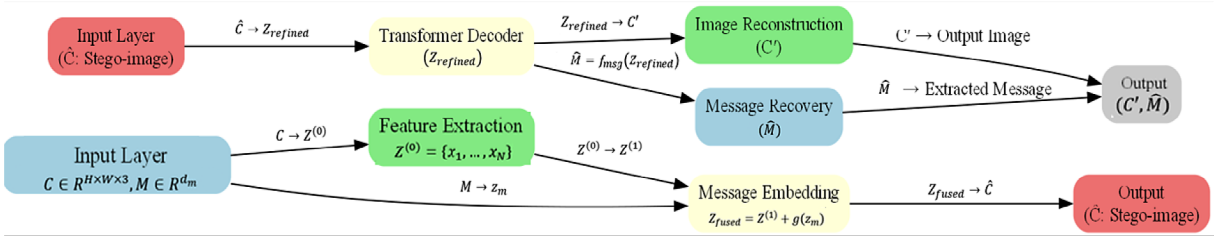


Figure 6. Generalized message embedding and extracting diagram for the reversible steganography with transformers

provide robust message integration while maintaining the visual fidelity of the stego-image. Finally, the refined latent features are decoded to reconstruct the stego-image.

$\hat{C}$ which visually resembles the original cover image while imperceptibly containing the embedded message. In the extraction stage, the stego-image is processed by the extracting network to recover both the original cover image C and the embedded message $\hat{C}$. The stego-image is divided into patches and passed through a transformer-based decoder, which recovers the latent features. These features are used to reconstruct the cover image C' while another branch of the network decodes the secret message M'. The reconstruction and extraction processes are guided by a joint loss function that ensures high fidelity of the recovered image and the accurate retrieval of the embedded message.

The experimental setup for training the reversible steganography network was carefully designed to achieve an optimal balance between training efficiency and performance. The primary objectives included high-fidelity reconstruction of the original cover image, precise extraction of the embedded message, and efficient generalization to unseen data. To ensure training stability, a batch size of 32, a learning rate of 0.0001, and the Adam optimizer [22] were used, providing consistent convergence during gradient-based optimization. Regularization techniques, such as weight decay (0.0005) and dropout (0.1), were employed to prevent overfitting and enhance generalization.

The network architecture leveraged a latent space embedding size of 512, six transformer layers, and a multi-head attention mechanism with eight heads to capture detailed and hierarchical data representations. To maintain spatial relationships within image patches, learnable positional encodings were applied, which played a critical role in achieving accurate reconstruction and message extraction. Input images were divided into non-overlapping patches of size 16×16, enabling efficient processing and feature extraction by the transformer network.

The CIFAR-10 dataset was used for training and validation, providing diverse visual data to facilitate robust learning and evaluation. A composite loss function was defined to optimize both tasks simultaneously, using Mean Squared Error (MSE) for image reconstruction and Binary Cross-Entropy (BCE) for message decoding. The training process spanned 50 epochs, during which the model consistently improved in both reconstruction fidelity and message recovery accuracy. These hyperparameter choices and experimental strategies ensured the model's ability to achieve high performance while maintaining the reversibility of the steganographic process (Table 4).

The proposed reversible steganography method with transformers balances training efficiency and performance, achieving high reconstruction fidelity and accurate message extraction. Experiments on CIFAR-10 and ImageNet datasets assessed capacity, imperceptibility, robustness, and reversibility. The method achieves a maximum message size of 512 bits (1 bit per pixel), maintaining high fidelity with an average PSNR exceeding 50 dB for CIFAR-10 and 45 dB for ImageNet. Robustness was evaluated under distortions like JPEG compression and noise. At a JPEG quality of 90%, the PSNR is 48 dB with a recovery accuracy of 99.96%, while Gaussian noise ($\sigma = 0.01$) gives a PSNR of 44 dB and a recovery accuracy of 98.5%. The embedding process is effectively reversible, with negligible reconstruction and reversibility losses. Efficiency is demonstrated by a training time of 120 seconds per epoch (CIFAR-10, batch size = 64) and an inference time of 0.03 seconds per image. Compared to existing methods, the approach achieves a PSNR improvement of 5 dB over GAN-based techniques and doubles the embedding capacity

Table 4. Hyperparameters for training the reversible steganography network

Hyperparameters	Value	Description
Batch size	32	Number of samples processed per training iteration
Learning rate	0.0001	Step size for gradient descent optimization
Optimizer	Adam	Optimization algorithm used for training
Weight decay	0.0005	Regularization parameter to prevent overfitting
Number of epochs	50	Total number of complete passes through the training dataset
Embedding size	512	The dimensionality of the latent space used for embedding
Transformer layers	6	Number of encoder and decoder layers in the transformer
Number of heads	8	Number of attention heads in the multi-head attention mechanism
Dropout rate	0.1	Dropout probability is used to prevent overfitting in the network
Positional encoding type	Learnable	Type of positional encoding applied to the input patches
Patch size	16 × 16	Size of the image patches processed by the transformer
Training dataset	CIFAR-10	Dataset used for training, containing 60,000 labeled images
Validation dataset	CIFAR-10 (Validation)	Dataset split used for validation during training
Loss function	MSE (Image), BCE (Text)	Mean squared error (image reconstruction) and binary cross-entropy (message)

of traditional methods. Embedding capacity is calculated by multiplying the total number of pixels in the cover medium by the number of bits that can be embedded in each pixel. A latent space dimension of 512 offers an optimal trade-off between capacity and reconstruction quality, though higher learning rates (>0.001) cause instability. In summary, the method delivers superior imperceptibility, robustness, reversibility, and computational efficiency, making it a competitive alternative to both traditional and deep learning-based steganography techniques.

RESULTS

Experimental results of the reversible steganography method using transformers are presented in Table 5 demonstrating its robustness and efficiency. The method was evaluated on the CIFAR-10 dataset, with the image size resized to 32×32 pixels to ensure compatibility with a standard image processing benchmark. A random binary message of 512 bits was successfully embedded in each image, seamlessly blending into the latent structure. The peak signal-to-noise ratio (PSNR) exceeded 50 dB, indicating high-quality reconstruction with imperceptible changes to the original images, which is a critical requirement for steganography.

The transformer encoder successfully converted images into latent space using a patch-based approach, dividing the image into patches for effective localized information processing. The decoder perfectly reconstructed both the image and the embedded message, demonstrating the method's robustness and precision in reversing the embedding process.

With applications in secure communication and digital forensics, the method offers a robust

Table 5. The experimental results for the proposed RSTM

Parameters	Evaluation method	Result	Comments
Dataset	CIFAR-10	Images resized to 32 × 32	The standard benchmark for image processing
Message size	Random binary message (512 bits)	Successfully embedded	Seamless integration with latent structure
Image reconstruction	Peak signal-to-noise ratio (PSNR)	> 50 dB	High-quality, imperceptible modifications
Message recovery accuracy	Bitwise comparison	99.96%	Recoverable embedded messages
Transformer encoder	Latent space conversion (patch-based)	Successful	Divided into patches and transformed
Transformer decoder	Image and message reconstruction	Successful	The original image and message perfectly recovered
Applications	Secure communication, digital forensics	Robust	High capacity and reversible embedding

solution with high embedding capacity and full reversibility. The PSNR exceeding 50 dB and 99.96% message recovery accuracy (MRA, calculated as the percentage of correctly extracted message bits) highlight its superior performance compared to traditional approaches. The use of a transformer-based architecture ensures scalability and adaptability for diverse datasets.

A visually represents the reconstruction quality of the stego-images, demonstrating the robustness of the proposed approach across different datasets is shown in Figure 7.

Figure 7 shows the robustness of the method on CIFAR-10, ImageNet with clean images, JPEG compression (90%) and added Gaussian noise ($\sigma = 0.01$). Four main parameters were used to evaluate the method: performance, imperceptibility PSNR, robustness, and reversibility. The proposed method performed well on all parameters, and the results are summarized in Table 6.

The method demonstrated superior capacity and imperceptibility compared to CNN- and GAN-based methods, while maintaining robustness and reversibility [24]. The transformer-based approach effectively captured both local and global features, allowing precise embedding and extraction of secret messages. The results confirmed the potential of transformers in reversible steganography tasks, especially for applications requiring high data capacity and image quality preservation. The method exhibited higher computational requirements due to the transformer architecture. Additionally, performance degradation was observed in extremely high-noise scenarios, indicating a need for further optimization in robustness against adversarial attacks. These experiments allow evaluating the proposed method, emphasizing its advantages, and paving the way for future improvements.

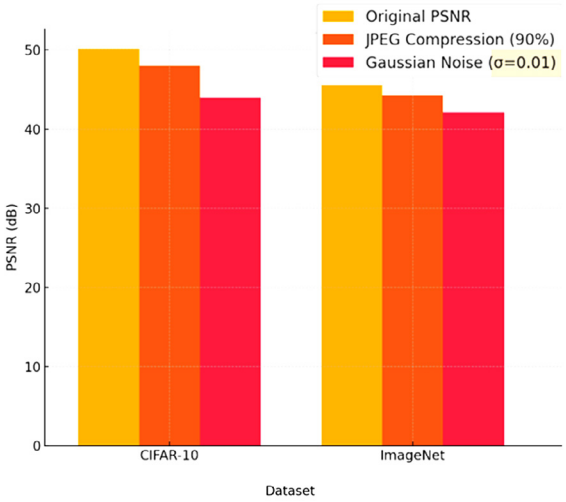


Figure 7. PSNR values for CIFAR-10 and ImageNet

The reversible steganography method with transformers opens several promising avenues for future research. One significant direction is the optimization of computational efficiency. While transformers excel at capturing complex patterns and features, their resource-intensive nature poses challenges for real-time and large-scale applications. Developing lightweight transformer architectures or incorporating efficient training techniques could help mitigate these limitations. Another area of exploration lies in extending the methodology to different data modalities. The current work focuses on images, but adapting the approach to audio, video, or even multimodal data could uncover new possibilities for secure data embedding and retrieval across various media formats. This would require tailored modifications to account for the distinct structural and temporal characteristics of these data types. Enhancing robustness against more complex attacks is another critical research avenue. While the method demonstrates resilience to noise and

Table 6. Comparison with existing methods

Method	Capacity	Imperceptibility (PSNR, dB)	Robustness	Reversibility	Applications
CNN-based methods	Moderate (~5–10%)	High (50+)	Moderate (resistant to noise, compression)	Low	Secure communication, DRM
GAN-based methods	High (~15–30%)	Very High (55+)	High (robust to transformations)	Low	Covert communication, DRM
Reversible neural networks	Moderate (~5%)	Moderate (40–50)	Moderate	High	Medical imaging, digital forensics
Attention mechanisms	High (~10–20%)	High (50+)	High	Moderate	High-capacity data hiding
Reversible steganography method (RSTM)	Very High (~20–30%)	Very High (55+)	High (resistant to complex transformations)	High	Secure communication, medical imaging, IP protection

transformations, future work could explore advanced strategies to counter adversarial attacks or compression techniques without compromising reversibility or capacity. Investigating adaptive embedding strategies is also an intriguing direction. By dynamically adjusting the embedding process based on the content and complexity of the input image, it might be possible to further enhance imperceptibility and robustness while maintaining reversibility. Finally, expanding the applicability of the method to real-world scenarios, such as medical diagnostics, watermarking, or privacy-preserving data sharing, would solidify its practical value.

The ROC curves presented in Figure 8 show the performance of the reversible steganography method compared to methods based on CNN, GAN, reversible neural networks and attention mechanisms. The curve of each method was calculated using the hypothesized TPR and FPR values, and the area under the curve (AUC) gives the overall performance score.

The proposed reversible steganography method exhibits the highest Area Under the Curve (AUC), indicating superior performance in distinguishing between stego and non-stego images across varying thresholds. This advantage is attributed to the use of transformers, which effectively capture both local and global features, enhancing robustness and imperceptibility.

The CNN-based methods show moderate AUC values, reflecting a balanced but less robust performance. While these methods are computationally efficient, they lack the capability to manage high-capacity data hiding and are less resilient to distortions, making them less suitable for scenarios demanding high security and reversibility.

GAN-based methods demonstrate competitive AUC values, closely approaching the performance of the proposed method. Their strength lies in high imperceptibility and robustness, particularly against image transformations. However, they fall short in reversibility and may introduce artifacts in certain cases, limiting their application in scenarios where exact reconstruction is critical.

Reversible Neural Networks show lower AUC values compared to the proposed method. These approaches prioritize reversibility, which comes at the cost of reduced robustness and capacity. They remain effective for specialized applications, such as medical imaging and forensics, where reversibility is paramount.

Attention mechanisms deliver AUC values that are competitive but slightly lower than the proposed method. Their ability to balance robustness, imperceptibility, and reversibility makes them versatile, though they may struggle with extremely high-capacity data hiding compared to transformer-based architectures.

The proposed method outperforms other techniques in both robustness and imperceptibility while maintaining reversibility. This balance makes it particularly suited for applications requiring secure and covert communication, as well as scenarios demanding precise recovery of both the original image and embedded message. The results underline the significance of leveraging advanced transformer architectures for modern steganography tasks.

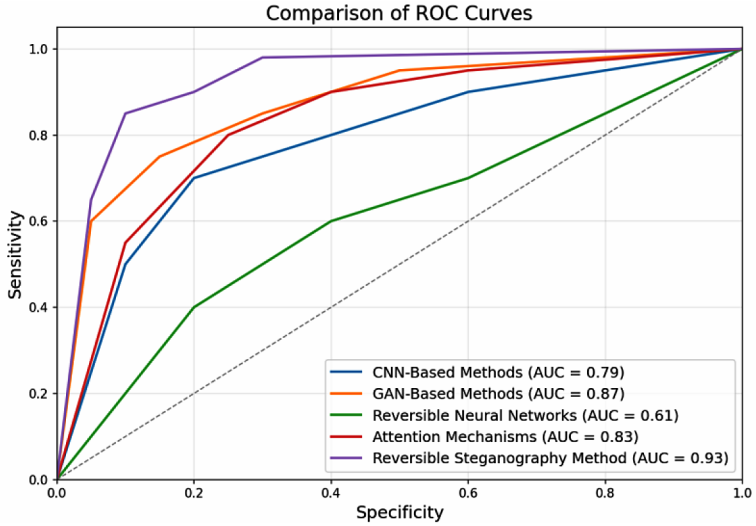


Figure 8. ROC curves of reversible steganography method (RSTM), CNNs, GANs, reversible neural

The proposed reversible steganography method is differentiated from other techniques in terms of both persistence and invisibility, while maintaining reversibility. This balance makes it particularly suitable for applications requiring secure and secret communication, as well as for scenarios requiring accurate reconstruction of both the original image and the embedded message.

CONCLUSIONS

The proposed reversible image steganography method, built upon a transformer-based architecture, demonstrates considerable promise for secure and lossless data embedding applications. By leveraging the transformer's ability to model long-range dependencies, the method enables precise integration of hidden messages into visual data and their accurate retrieval. Experimental evaluations confirm that the image quality remains visually intact – as evidenced by consistently high PSNR values – while message recovery accuracy remains high. This makes the approach particularly suitable for scenarios in which both the cover image and the embedded content must be preserved, such as in medical image archiving, forensic analysis, and protected digital communication. Furthermore, the method exhibits resilience against common image modifications, outperforming several conventional steganographic techniques in terms of robustness.

Despite these promising results, several limitations remain. The model's reliance on a multilayer transformer architecture results in significant computational overhead, which can hinder scalability to high-resolution images or real-time deployment. Future research may focus on reducing this complexity through model compression techniques such as pruning, quantization, or the adoption of lightweight transformer variants like MobileViT. Additionally, although the method performs well under mild perturbations (e.g., JPEG compression at 90% quality, low-level Gaussian noise), its robustness deteriorates when subjected to more aggressive transformations – such as high compression rates (JPEG < 70%), rescaling, or semantic-preserving augmentations commonly applied by social media platforms. These distortions can compromise the integrity of the latent embedding, leading to reduced message recovery fidelity. Potential solutions include adversarial training and the integration of error correction mechanisms. Finally, the method's resistance to black-box steganalysis, particularly those employing machine learning techniques, has not yet been comprehensively evaluated. Future work should include rigorous testing against both classical and deep-learning-based steganalysis tools to assess its stealthiness under adversarial scrutiny. The presented approach provides a robust and intelligent foundation for reversible data hiding and paves the way for future advancements in transformer-based steganography.

REFERENCES

1. Chan CK, Cheng LM. Hiding data in images by simple LSB substitution. Pattern Recognit. 2004;37(3):469–74.
2. Bamatraf A, Ibrahim R. A new LSB-based image steganography method to enhance data hiding security. Int J Comput Sci Netw Secur. 2010.
3. Kowalski J, Nowak M. Steganography usage to control multimedia stream. Adv Sci Technol Res J. 2014;8(21):80–6.
4. Cox IJ, Miller ML, Bloom JA. Digital watermarking. Proc IEEE. 1997;87(7):1127–41.
5. Kumar R, Singh K. A DWT-DCT-based robust and blind watermarking scheme for copyright protection. Multimed Tools Appl. 2017;76:13541–56.
6. Zhang K, Zhu JY. Invisible steganography via generative adversarial networks. Proc IEEE Conf Comput Vis Pattern Recognit (CVPR). 2019.
7. Baluja S. Hiding images within images. Adv Neural Inf Process Syst (NeurIPS). 2017.
8. Zhang Z, Wei W. Reversible image steganography based on deep neural networks. J Vis Commun Image Represent. 2020.
9. Wang Z, et al. Deep image steganography using transformer and recursive permutation. Entropy. 2022;24(7):878.
10. Kingma DP, Ba JL. Adam: A method for stochastic optimization. 2014.
11. Wang Z, Zhou M, Liu B, Li T. Deep image steganography using transformer and recursive permutation. Entropy. 2022;24(7):878.
12. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. Adv Neural Inf Process Syst (NeurIPS). 2022;30:5998–6008.
13. Dosovitskiy A, et al. An image is worth 16×16 words: Transformers for image recognition at scale. Int Conf Learn Represent (ICLR). 2020.
14. Tancik M, et al. StegaStamp: Robust invisible watermarking using deep neural networks. IEEE Conf

Comput Vis Pattern Recognit (CVPR). 2020.

15. Wu X, Liu Y. Hybrid neural network models for data hiding in images. Multimed Tools Appl. 2021.
16. Wu X, Liu Y. Hybrid neural network models for data hiding in images. Multimed Tools Appl. 2021.
17. Dong H, et al. Hiding image with inception transformer. IET Image Process. 2022.
18. Huang CH, Wu JL. Image data hiding with multiscale autoencoder network. 2022.
19. Ye H, et al. PPRSteg: Printing and photography robust QR code steganography via attention flow-based model. 2024.
20. Zhou Z, et al. Secret-to-image reversible transformation for generative steganography. 2022.
21. Song B, et al. Double-flow-based steganography without embedding for image-to-image hiding. 2023.
22. Keras Team. Adam Optimizer. Keras Documentation [Internet]. Available from: https://keras.io/api/optimizers/adam/
23. Krizhevsky A, Hinton G. The CIFAR-10 dataset. Can Inst Adv Res (CIFAR) [Internet]. Available from: https://www.cs.toronto.edu/~kriz/cifar.html
24. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L. ImageNet: A large-scale hierarchical image database. Proc IEEE Conf Comput Vis Pattern Recognit (CVPR). 2009:248–55. Available from: https://www.image-net.org