Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

SCFformer: a binary data hiding method against JPEG compression based on spatial channel fusion Transformer*

Xintao DUAN†‡[1,2], Chun LI[1,2], Bingxin WEI[3], Guoming WU[1,2], Chuan QIN[4], Haewoon NAM[3]

[1]*School of Computer and Information Engineering, Henan Normal University, Xinxiang 453007, China*

[2]*Key Laboratory of Artificial Intelligence, Henan Normal University, Xinxiang 453007, China*

[3]*Department of Electrical and Electronic Engineering, Hanyang University, Ansan 15588, Republic of Korea*

[4]*School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China*

†E-mail: duanxintao@htu.edu.cn

Received Nov. 8, 2023; Revision accepted Mar. 28, 2024; Crosschecked Mar. 13, 2025

Abstract: To enhance information security during transmission over public channels, images are frequently employed for binary data hiding. Nonetheless, data are vulnerable to distortion due to Joint Photographic Experts Group (JPEG) compression, leading to challenges in recovering the original binary data. Addressing this issue, this paper introduces a pioneering method for binary data hiding that leverages a combined spatial and channel attention Transformer, termed SCFformer, to withstand JPEG compression. This method employs a novel discrete cosine transform (DCT) quantization truncation mechanism during the hiding phase to bolster the stego image's resistance to JPEG compression, using spatial and channel attention to conceal information in less perceptible areas, thereby enhancing the model's resistance to steganalysis. In the extraction phase, the DCT quantization minimizes secret image loss during compression, facilitating easier information retrieval. The incorporation of scalable modules adds flexibility, allowing for variable-capacity data hiding. Experimental findings validate the high security, large capacity, and high flexibility of our scheme, alongside a marked improvement in binary data recovery post-JPEG compression, underscoring our method's leading-edge performance.

Key words: Binary data hiding; Against JPEG compression; Discrete cosine transform quantization; SCFformer
https://doi.org/10.1631/FITEE.2300762 **CLC number:** TP309.7

1 Introduction

As a method of embedding secret information in digital media, data hiding technology (Zhu et al., 2018; Li S and Zhang, 2019; Shang et al., 2023) has important application value in protecting privacy and confidential communication. In the field of data hiding, images (Mallika et al., 2022), text (Jassim, 2013), video (Mou et al., 2023), and speech signals (Abood et al., 2022) can be transmitted as message carriers. Images have been widely used as carriers of binary information due to their high redundancy of information. However, in practical application scenarios, social platforms and major websites use compression operations on images due to storage and transmission costs, the most common of which is Joint Photographic Experts Group (JPEG) compression. After the image is compressed, the binary information cannot be completely extracted, and the non-difference of JPEG makes it more difficult to train a model to resist lossy compression. This affects the significance of the user for private

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (Nos. U1904123, 62172280, and U20B2051), the Key Scientific Research Projects of Colleges and Universities in Henan Province, China (No. 23A520006), and the Henan Provincial Science and Technology Research Project, China (No. 222102210199)

ORCID: Xintao DUAN, https://orcid.org/0000-0001-8757-2447

transmission of information. The essence of transmission should be to ensure the integrity of information. Therefore, a data hiding algorithm with practical application value should ensure not only strong anti-detection ability, but also the complete extraction of embedded messages.

The algorithm proposed by Tao et al. (2019) can resist the content loss caused by JPEG compression. However, this method uses an ideal JPEG channel, which may not be applied to non-differential JPEG compression channels in practical applications. Therefore, Lu et al. (2021) proposed a JPEG data hiding algorithm, which can resist JPEG compression with quality factor (QF) of 75, 85, and 95, and use adaptive Bose–Chaudhuri–Hocquenghem (BCH) coding to decrease the error rate (ER) of secret message extraction. However, these algorithms still cannot resist JPEG compression with QF lower than 75 in the face of high capacity, and may not meet the requirements of current social networks. Shang et al. (2023) directly embedded the secret message into the quantized discrete cosine transform (DCT) coefficients of the cover image and decoded the secret message from the quantized DCT coefficients of the attacked stego image. Although this method has a certain resistance to lossy JPEG compression in essence, the extraction ER of the extracted secret message is still not ideal. Kishore et al. (2022) achieved low extraction ER by simulating the JPEG compression process, and the algorithm can resist JPEG compression with QF = 80. However, these methods lack effectiveness in practical applications.

Targeting the inadequacy of current data hiding algorithms, which exhibit significant extraction errors after low-quality-factor JPEG compression and demonstrate limited practical applicability, a novel anti-JPEG compression binary data hiding method based on spatial channel fusion Transformer (SCFformer) is proposed, which can recover secret information with high precision after image processing. First of all, we use an efficient fusion Transformer to capture the dimensional relationship of the entire space and channels while maintaining computational efficiency. Second, we introduce a DCT quantization truncation mechanism to improve the anti-JPEG compression ability. In addition, we use a scalable module to realize the flexibility of secret information hiding, so that our scheme can deal with different capacities of secret information. The specific contributions are as follows:

1. A novel anti-JPEG compression binary data hiding method based on SCFformer is proposed, which can provide high-precision secret information recovery with low QF.

2. A DCT quantization truncation mechanism is proposed, which improves the anti-JPEG compression ability of the stego image, improves the flexibility by using the scalable module, and realizes secret data hiding with variable capacity.

3. An efficient fusion Transformer is proposed, which improves the ability of capturing the dimensional relationships of the entire space and channels while maintaining high efficiency, thereby achieving a lower secret data extraction ER at the same capacity.

2 Related works

2.1 Traditional anti-JPEG compression data hiding

Since the emergence of image watermarking, researchers have explored various methods to improve the anti-JPEG compression ability and hiding capacity of data hiding. To improve the anti-JPEG compression ability of data hiding, the trend has shifted to hiding data in the transform domain, such as DCT (Singh and Bhatnagar, 2018), discrete wavelet transform (Lu et al., 2009), or a mixture of them (Singh et al., 2012; Harish et al., 2013). To achieve a good balance between hiding capacity and anti-JPEG compression capability, adaptive data hiding has emerged to embed hidden data based on the local features of the carrier image (Bhinder et al., 2018; Chen et al., 2018). In Zhang Y et al. (2015), by using the relative relationship between DCT coefficients, a message embedding method with strong resistance to JPEG compression was constructed. This method can resist JPEG compression under multiple QFs, but it will sacrifice some anti-detection capabilities under high embedding capacity. Yu et al. (2020) proposed an anti-JPEG data hiding method combined with jitter modulation, which constructs the asymmetric cost of different modification polarities. By combining the recompression knowledge, this method can improve the anti-JPEG compression ability and security under the known channel QF. In Zhang Y et al. (2021), by analyzing the principle of

JPEG compression, the quantization rounding principle was used to construct the message embedding domain, and based on the coefficient relationship and adaptive modification, it has the ability to resist JPEG compression and Gaussian noise. In addition, the method combines the embedding cost calculation function and the coding method to achieve message embedding, which achieves higher embedding efficiency. In Zhao et al. (2019), an anti-JPEG compression method based on transmission channel matching was proposed. This method adjusts the carrier image by repeatedly inputting the carrier image into the transmission channel until the generated carrier image can resist the JPEG compression of the transmission channel. In addition, they used adaptive BCH coding to reduce the extraction ER. However, these methods cannot solve the problem of third-party interference in public channels faced by data hiding.

2.2 Anti-JPEG compression data hiding based on deep learning

Traditional data hiding methods usually require practitioners to have a good understanding of a wide range of related professional knowledge, and methods based on deep learning automate the entire process, which greatly promotes their widespread use. In addition to its success in a wide range of applications, deep learning has shown a trend of success in data hiding (Isac and Santhi, 2011). Shang et al. (2023) have shown the possibility of hiding another complete image in one image. In Hayes and Danezis (2017), a method of hiding binary data by adversarial learning was proposed. The above deep learning-based methods do not take into account their anti-interference ability, and the hidden data can be easily destroyed by some common lossy compression methods such as JPEG. To improve the resistance to JPEG compression, Zhu et al. (2018) proposed to add a "noise layer" to simulate JPEG during the training process to enhance the encoded image. Since the standard JPEG pipeline contains non-differentiable quantization steps, researchers developed differentiable surrogates like JPEG-Mask and JPEG-Drop. These methods emulate JPEG's frequency-domain processing by controllably discarding high-frequency DCT coefficients during gradient propagation. Although JPEG-Mask and JPEG-Drop remove high-frequency content like real JPEG, their behavior is still far from that of real JPEG, resulting in overfitting and poor performance when tested under real JPEG compression. To minimize the gap between approximate and "real" JPEG, a new method was proposed by Li ZZ et al. (2023) to carefully simulate the important steps in "real" JPEG. A similar method was proposed by Bui et al. (2023) to generate adversarial samples against JPEG compression. This paper uses a similar method to combat JPEG compression.

3 Proposed method

3.1 Net framework

We introduce a non-convolutional U-Net-like SCFformer data hiding scheme, as shown in Fig. 1. In the hiding process, we have the cover image $\boldsymbol{X}_{\mathrm{co}} \in \mathbb{R}^{H\times W\times C}$ and the secret information $\boldsymbol{X}_{\mathrm{se}} \in \mathbb{R}^{\frac{H}{4}\times\frac{W}{4}\times n}$. The spatial dimension of the image is

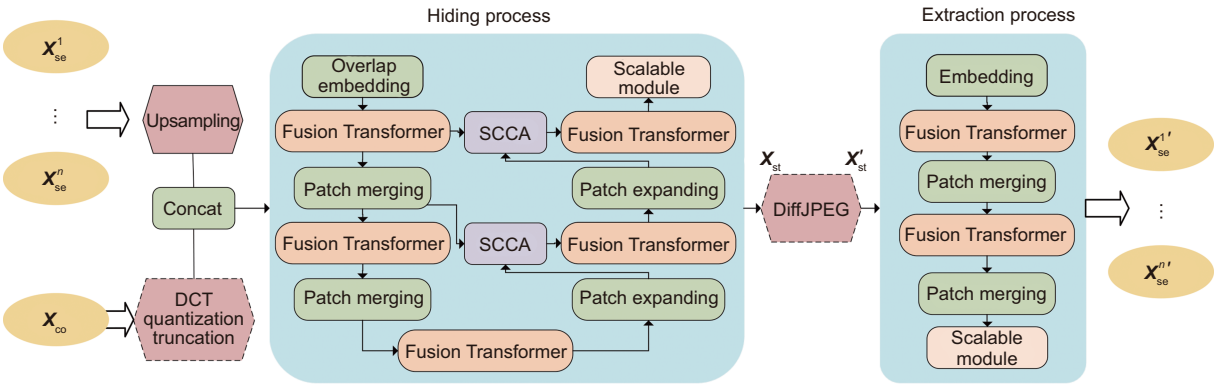

Fig. 1 The overall framework of SCFformer data hiding scheme including mainly two parts, hiding process and extraction process

$H \times W$, the number of channels of the image is C, the spatial dimension of the secret information is $\frac{H}{4} \times \frac{W}{4}$, and the number of channels of the secret information is n. First, $\boldsymbol{X}_{\text{se}}$ is upsampled and merged with the DCT quantization truncation cover image $\boldsymbol{X}_{\text{co}}$. Then we use the patch merging module (Liu et al., 2021; Huang et al., 2023) to obtain 4×4 tokens of size from the merged results. Finally, the input will pass through the encoder module, which has two stacked encoder blocks, each of which consists of a fusion Transformer layer and a patch merging layer. The decoder module, on the other hand, consists of a fusion Transformer layer and a patch expanding layer. During hierarchical token merging, each 2×2 local window undergoes spatial downsampling (stride=2) with concomitant channel doubling via depthwise concatenation. This parametric scaling mechanism enables progressive multi-resolution representation learning across Transformer stages. In the decoder, the tokens in each block are expanded twice again. Then, the output of each patch merging layer is fused with the skip connection of the parallel encoder layer using skip connection cross attention (SCCA).

3.2 Efficient fusion Transformer

The traditional self-attention mechanism generates a redundant context matrix, and Eq. (1) is a more efficient process of calculating the self-attention mechanism:

$$\boldsymbol{E}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \rho_q(\boldsymbol{Q})(\rho_k(\boldsymbol{K}^{\mathrm{T}})\boldsymbol{V}), \tag{1}$$

where the query, key, and value matrices $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V} \in \mathbb{R}^{n' \times d}$ are projected into a low-dimensional space via low-rank projection functions ρ_q and ρ_k (e.g., $\rho_q(\boldsymbol{Q}) = \boldsymbol{Q}\boldsymbol{W}_q$ with $\boldsymbol{W}_q \in \mathbb{R}^{d \times k}$, where $k \ll n'$). This avoids the explicit computation of the $n' \times n'$ matrix in traditional self-attention $\mathrm{Softmax}(\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}})$, reducing the computational complexity from $O(n'^2 d)$ to $O(n'kd)$.

The attention mechanism can help the network pay more attention to the unimportant parts of the carrier when embedding secret data, so as to reduce the impact on the stego image as much as possible and improve the concealment and security of the embedded secret data. We consider that different types of attention mechanisms can be used to achieve the above goals. To capture cross-channel information more effectively, we use the transposed attention mechanism. Eq. (2) shows the process of the transposed attention mechanism:

$$\begin{cases} \boldsymbol{T}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \boldsymbol{V}\boldsymbol{C}_{\mathrm{T}}(\boldsymbol{K}, \boldsymbol{Q}), \\ \boldsymbol{C}_{\mathrm{T}}(\boldsymbol{K}, \boldsymbol{Q}) = \mathrm{Softmax}\left(\dfrac{\boldsymbol{K}^{\mathrm{T}}\boldsymbol{Q}}{\tau}\right), \end{cases} \tag{2}$$

where τ represents the temperature parameter, which is used to prevent overfitting.

After $\boldsymbol{Q}$ and $\boldsymbol{K}$ are transposed, the attention weight must be calculated based on the cross-covariance matrix. $\boldsymbol{C}_{\mathrm{T}}(\boldsymbol{K}, \boldsymbol{Q})$ is a global context vector that refers to the transposition of attention. Before calculating the attention weight, the temperature parameter τ is introduced to offset the L_2-norm scaling from $\boldsymbol{Q}$ and $\boldsymbol{K}$. This operation will improve the stability of the training process, but reduce a certain degree of freedom. The literature on attention mechanism (Guo MH et al., 2022) shows that the combination of spatial attention and channel attention can improve the ability to capture more contextual features. Therefore, we construct an efficient fusion Transformer which combines transpose (channel) attention and efficient (spatial) attention. Its structure is shown in Fig. 2. It includes an efficient attention block, followed by an addition and normalization block, a transposed attention block, and finally an addition and normalization block, which is expressed in Eq. (3). The structure can capture more unnoticed image regions efficiently, thereby efficiently hiding information.

$$\begin{cases} \boldsymbol{E}_{\text{block}}(\boldsymbol{X}, \boldsymbol{Q}_1, \boldsymbol{K}_1, \boldsymbol{V}_1) = \boldsymbol{E}(\boldsymbol{Q}_1, \boldsymbol{K}_1, \boldsymbol{V}_1) + \boldsymbol{X}, \\ \mathrm{MF}_1(\boldsymbol{E}_{\text{block}}) = \mathrm{MF}(\mathrm{LN}(\boldsymbol{E}_{\text{block}})), \\ \boldsymbol{T}_{\text{block}}(\boldsymbol{E}_{\text{block}}, \boldsymbol{Q}_2, \boldsymbol{K}_2, \boldsymbol{V}_2) \\ \quad = \boldsymbol{T}(\mathrm{MF}_1(\boldsymbol{E}_{\text{block}}) + \boldsymbol{E}_{\text{block}}) + \mathrm{MF}_1(\boldsymbol{E}_{\text{block}}), \\ \mathrm{MF}_2(\boldsymbol{T}_{\text{block}}) = \mathrm{MF}(\mathrm{LN}(\boldsymbol{T}_{\text{block}})), \\ \mathrm{FT}(\boldsymbol{T}_{\text{block}}) = \mathrm{MF}_2(\boldsymbol{T}_{\text{block}}) + \boldsymbol{T}_{\text{block}}. \end{cases} \tag{3}$$

Here, $\boldsymbol{T}_{\text{block}}$ denotes the transposed attention block and $\boldsymbol{E}_{\text{block}}$ denotes the spatial attention block. $\boldsymbol{Q}_1$, $\boldsymbol{K}_1$, and $\boldsymbol{V}_1$ are calculated based on the input $\boldsymbol{X}$, and $\boldsymbol{Q}_2$, $\boldsymbol{K}_2$, and $\boldsymbol{V}_2$ are calculated from the input of the transposed attention block. LN represents regularization and FT represents fused Transformer. MF represents the mix-FFN (FFN is short

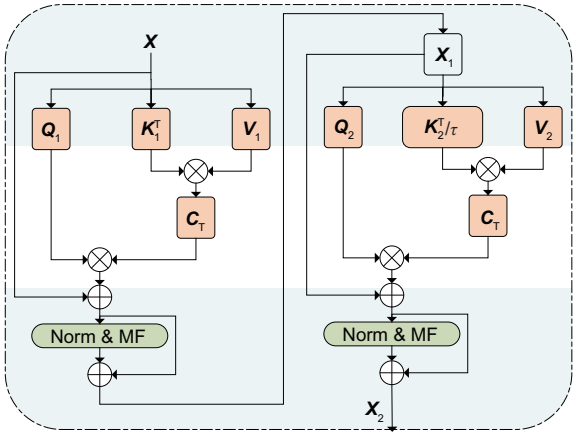

Fig. 2 Efficient fusion Transformer module including an efficient spatial attention block and a channel attention block, followed by a specification and mix-feedforward (MF) neural network

for feedforward neural network) (Huang et al., 2023):

$$\mathrm{MF}(\boldsymbol{X}) = \mathrm{FC}(\mathrm{GELU}(\mathrm{DW}(\mathrm{FC}(\boldsymbol{X})))), \quad (4)$$

where FC is a fully connected layer, GELU represents the Gaussian error linear unit activation, and DW represents depthwise convolution.

3.3 Skip connection cross attention

The SCCA module in Fig. 3 establishes bidirectional feature dialogues between encoder–decoder pairs. The module can effectively provide spatial information for each encoder, so that fine-grained information can be restored when a stego image is generated. In Eq. (5), SCCA applies efficient attention, but it is not simple to use the same input for $\boldsymbol{K}$, $\boldsymbol{Q}$, and $\boldsymbol{V}$. $\boldsymbol{Q}$ is output by the encoder layer of $\boldsymbol{X}_2$, and the input of $\boldsymbol{K}$ and $\boldsymbol{V}$ is the output of the decoder layer $\boldsymbol{X}_1$. To fuse these two features, $\boldsymbol{X}_1$ needs to be scaled to the same dimension as $\boldsymbol{X}_2$ through a linear layer for embedding. The role behind using $\boldsymbol{X}_2$ as $\boldsymbol{Q}$ input is to perform multi-level representation in efficient spatial attention blocks.

$$\begin{cases} \boldsymbol{K}, \boldsymbol{V} = \mathrm{Proj}(\mathrm{FC}(\boldsymbol{X}_1)), \\ \boldsymbol{Q} = \mathrm{Proj}(\boldsymbol{X}_2), \\ \boldsymbol{E} = \rho_v(\boldsymbol{V})\rho_k(\boldsymbol{K}^{\mathrm{T}})\rho_q(\boldsymbol{Q}). \end{cases} \quad (5)$$

3.4 DCT quantization truncation mechanism

In JPEG compression, there is a special mechanism called recompression. In the implementation

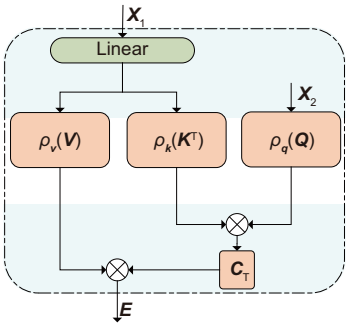

Fig. 3 The SCCA module containing a linear layer and an efficient attention module

of some JPEG data hiding schemes (such as OutGuess and F5), additional JPEG compression may be performed before data embedding, and then in the social network, the picture will be recompressed and the DCT coefficient of the stego image will be changed, resulting in adaptive data hiding failure. Although some methods can hide information in the DCT frequency domain, they still cause the data hiding scheme to fail when the image is uploaded to an online social network. We find that we can effectively deal with the problems caused by JPEG compression by quantifying an image by DCT quantization, generating the corresponding DCT quantization coefficients, performing quantization truncation according to the degree of interference (that is, the JPEG quality compression factor in the disturbed scene), performing inverse DCT transformation, and finally performing data embedding. This mechanism is called DCT quantization truncation. Under this mechanism, we can select a set of specific DCT quantization tables and use them to convert the original image into DCT coefficients. Then, we can embed these coefficients without worrying about the changes of DCT coefficients that may be caused by JPEG compression. After embedding the data, we can restore the DCT coefficients to the image as needed. This mechanism can improve the anti-JPEG compression ability of the data hiding scheme, and can realize secure embedding and extraction of data without affecting the image quality. This mechanism has been widely used in various JPEG data hiding schemes and proved to be a very effective solution.

3.5 Key control mechanism

This paper introduces a new key control scheme, which can be used to further encrypt the secret information when the image is hidden and to recover the

secret information under certain key conditions. The design structure of this scheme is shown in Fig. 4. In this scheme, given a cover image, it first passes through a fusion Transformer layer. Next, a specific key is generated by a key encoder, which consists of an FC layer. Then, the input key at the end of the fusion transformer layer is used to generate the condition vectors of the two channels, which are divided into two modulation vectors in the perspective of channel dimension. During the training process, we constrain the output key and secret information during extraction. During the training process, the scheme constrains the output, key, and secret information during extraction. In this way, it can ensure that the scheme has higher security and accuracy when hiding and restoring secret information.

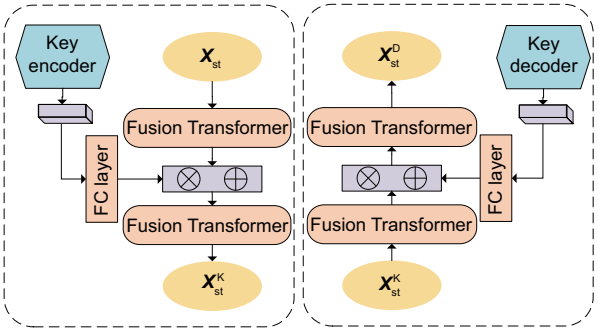

Fig. 4 The key control mechanism consisting of two parts, key generation and information extraction

3.6 Scalable module

Our scalable design, shown in Fig. 5, can be considered as a special convolution layer whose kernel size can be changed according to the input. All convolution kernels of different dimensions come from the same base kernel and share parameters. For a given input feature, we intercept a convolution kernel with the size of $k\times k$ to match the input dimension and then perform convolution. We implement this operation by training all convolution kernels.

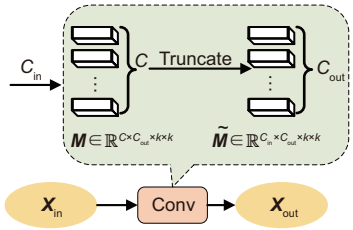

Fig. 5 Scalable module

3.7 Loss function and training process

To understand the influence of different scenarios on the data hiding effect, we use three model variants to calculate the loss function. For convenience, three different variants are called variation A, variation B, and variation C. To distinguish different variants, we set different security levels. Variation A can be used in scenarios where the user requires higher security without considering the computational cost, variation B can be used in scenarios where the user requires certain security and considers the computational cost, and variation C can be used in scenarios where the user requires certain security and considers the image quality. In our model, the loss function is used to constrain two parts, namely hiding and recovering. Hiding is to hide multiple secret messages in the carrier image. The generated stego image $\boldsymbol{X}_{\rm st}$ should prevent detectable traces of the embedded secret information while maintaining maximum visual similarity to the original carrier image. Therefore, we constrain the stego image $\boldsymbol{X}_{\rm st}$ to be the same as the cover image $\boldsymbol{X}_{\rm co}$. In variation A, we use the DCT quantization truncation mechanism to calculate the hidden loss:

$$\mathcal{L}_{\rm h}=\frac{1}{N}\sum_{i=1}^{N}\left\|\text{DiffJPEG}(\boldsymbol{X}^i_{\rm st},\text{QF})-\boldsymbol{Q}_T(\boldsymbol{X}^i_{\rm co},\text{QF})\right\|_2^2, \tag{6}$$

where N denotes the batch size, DiffJPEG(,) simulates differentiable JPEG compression, $\boldsymbol{Q}_T(,)$ applies DCT-based quantization with table T, and QF $\in[1,100]$ controls the compression strength, ensuring frequency-domain alignment between $\boldsymbol{X}_{\rm st}$ and $\boldsymbol{X}_{\rm co}$ under JPEG transformation.

In variation B, we abandon the DCT quantization truncation mechanism to calculate the hidden loss:

$$\mathcal{L}_{\rm h}=\frac{1}{N}\sum_{i=1}^{N}\left\|\boldsymbol{X}^i_{\rm co}-\boldsymbol{X}^i_{\rm st}\right\|_2^2. \tag{7}$$

In variation C, based on the DCT quantization truncation mechanism, we perform the inverse process of image degradation on the generated carrier image, trying to restore the image degradation caused by JPEG compression, and calculate the hidden loss:

$$\mathcal{L}_{\rm h}=\frac{1}{N}\sum_{i=1}^{N}\left\|\text{DiffJPEG}(\boldsymbol{X}^i_{\rm st},\text{JCF})-\boldsymbol{X}^i_{\rm co}\right\|_2^2, \tag{8}$$

where JCF represents the JPEG compression factor. In the recovery process, our goal is to recover the secret information from the stego image to be the same as the original secret information. Then the corresponding recovering loss function is shown in Eq. (9):

$$\mathcal{L}_{\mathrm{r}}=\frac{1}{N}\sum_{i=1,j=1}^{N}\left\|\hat{\boldsymbol{X}}_{\mathrm{se},j}^{i}-\boldsymbol{X}_{\mathrm{se},j}^{i}\right\|_{2}^{2}, \tag{9}$$

where $\boldsymbol{X}_{\mathrm{se},j}^{i}$ and $\hat{\boldsymbol{X}}_{\mathrm{se},j}^{i}$ are the i^{th} secret message and the i^{th} extracted secret message of the j^{th} batch, respectively.

For the overall optimization model, we calculate the total loss by minimizing the hidden loss and the recovering loss:

$$\mathcal{L}=\mathcal{L}_{\mathrm{h}}+\lambda\mathcal{L}_{\mathrm{r}}, \tag{10}$$

where λ is a hyperparameter that can be traded between hiding and recovering. We set $\lambda=0.5$ to balance these two losses. The whole training process of this method is shown in Algorithm 1, where θ denotes the parameter of the model and t is the number of iterations.

Algorithm 1 Training process

Input: Secret messages $\boldsymbol{X}_{\mathrm{se}}^{1},\boldsymbol{X}_{\mathrm{se}}^{2},\ldots,\boldsymbol{X}_{\mathrm{se}}^{n}$; cover image $\boldsymbol{X}_{\mathrm{co}}$
Output: Trained networks Hiding(), Reveal(); stego image $\boldsymbol{X}_{\mathrm{st}}$; recovered messages $\boldsymbol{X}_{\mathrm{se}}^{1\prime},\boldsymbol{X}_{\mathrm{se}}^{2\prime},\ldots,\boldsymbol{X}_{\mathrm{se}}^{n\prime}$

```
Initialize all layers
for each epoch do
    for each iteration do
        X_se ← Concat(X_se^1, X_se^2, ..., X_se^n)
        X'_co ← Q_T(X_co)
        X_st ← Hiding(X_se, X'_co)
        Loss ← MSE(X_st, X'_co)
        X'_st ← DiffJPEG(X_st)
        X_se^1', X_se^2', ..., X_se^n' ← Reveal(X'_st)
        Update parameters via the gradient descent method:
            θ_t ← θ_t − η ∂L/∂θ_t
    end for
    Shuffle the training set
end for
```

4 Experiments

This section mainly introduces the experimental situation of our work. We first give an overview of the implementation details, briefly introduce the evaluation indicators used, and then perform capacity analysis, visual quality analysis, anti-JPEG compression capability analysis, and security analysis. Finally, ablation experiments are carried out to verify the effectiveness of the proposed method.

4.1 Experimental setting

In this work, we use 30 000 images selected from ImageNet as the cover images of the training set to train our model. The spatial resolution is 224×224. We use AdamW (Zhang Y et al., 2021) as the optimizer. $\beta_1=0.9$ and $\beta_2=0.5$. We set the batch size to 16. The weight attenuation factor is set to 1×10^{-12}. Using an initial learning rate of 1×10^{-4}, the hidden network will be reduced by half every 5 iterations without decreasing the loss, and the recovery network will be reduced by half every 10 iterations without decreasing the loss. The total number of iterations is set to 300. On an NVIDIA Tesla V100 Graphics Processing Unit (GPU), the training process can be completed within 5 days.

4.2 Parameter evaluation

In this paper, the error-sensitive peak signal-to-noise ratio (PSNR) (Horé and Ziou, 2010) and structural similarity index measurement (SSIM) (Ye YX et al., 2017) are used as indicators to measure the data hiding effect. MSE represents the mean square error between the cover image $\boldsymbol{X}_{\mathrm{co}}$ and the stego image $\boldsymbol{X}_{\mathrm{st}}$. PSNR has been widely used in the field of computer vision. Its calculation source is the difference between the pixels at the same position in the two images, which can objectively reflect the image quality after compression or other processing. Since the image often deviates during transmission or processing, the higher the PSNR value, the smaller the image change after processing. In this paper, PSNR is selected for measurement because it can accurately measure the difference between the cover image and the stego image. In this paper, the embedding of information in the steganography process is regarded as the destruction of the image quality, so PSNR can be used to evaluate the distortion degree of the secret information embedded in the stego image. P_{e} is a commonly used measure to evaluate the undetectability of the stego image:

$$P_{\mathrm{e}}=\frac{1}{2}\left(P_{\mathrm{FA}}+P_{\mathrm{MD}}\right), \tag{11}$$

where P_{FA} and P_{MD} are the false positive rate and missed detection rate, respectively. The value

range of P_e is [0, 1], and the optimal value is 0.5. Data hiding capacity can be obtained by Payload by calculating the number of bits per pixel:

$$\text{Payload} = \frac{m}{W \times H}, \tag{12}$$

where m represents the number of bits of the embedded binary data, and W and H represent the width and height of the image embedded in the data, respectively.

ER can be calculated by Eq. (13), where $\boldsymbol{X}_{\text{se}}^{i}$ represents the i^{th} secret message to be sent, $\boldsymbol{X}_{\text{se}}^{i\prime}$ represents the recovered secret message to be sent, and N_s denotes the number of bits of secret messages. After our algorithm is embedded into the cover image, the cover image is compressed and interfered, and the secret message is re-extracted.

$$\text{ER} = \sum_{i=1}^{N_s} \left(\left| \boldsymbol{X}_{\text{se}}^{i} - \boldsymbol{X}_{\text{se}}^{i\prime} \right| \Big/ m \right). \tag{13}$$

4.3 Capacity analysis

Effective data hiding algorithms need to consider three key factors: visual quality, security, and payload capacity. Payload capacity refers to the amount of data that can be hidden under the premise of maintaining the visual quality of the image. If the payload capacity is larger, the visual quality of the image will become lower, and vice versa. Therefore, a well-designed data hiding algorithm should have a high payload capacity while maintaining the visual quality of the image. The values shown in Table 1 are calculated according to Eq. (12), where n, the number of channels of the secret information, is the capacity criterion of our method.

4.4 Visual quality analysis

Visual quality is an important metric for data hiding algorithms and an important consideration for user selection algorithms. When $n = 3$, the subjective visual quality of our algorithm under different QFs is shown in Fig. 6. When QF=75, the subjective visual quality under different n's is shown in Fig. 7. It can be seen from the graph that under different QFs, the stego image and the stego image after interference still maintain visual similarity to the original image, and the residual is basically invisible. From the histogram of the residual graph, it can be seen that the residual pixels are basically concentrated in the left region to ensure the concealment of the message. Therefore, in practical applications, objective assessment is widely used. In general, we use PSNR and SSIM to evaluate the impact of data hiding on the image, so as to obtain the visual quality of the image. Generally speaking, the higher the PSNR and SSIM, the less the impact of the data hiding process on the image, and the better the visual quality of the image. We draw the trend of SSIM and PSNR under different n values and different QF values. It can be seen from Fig. 8 that as the capacity increases, the image quality continues to decline, but it is still within the acceptable range. Under the same capacity, the higher the QF value, the better the visual performance.

Table 1 Capacity analysis with different values of n

n	Payload	n	Payload
1	0.0625	5	0.3125
2	0.1250	6	0.3750
3	0.1875	7	0.4375
4	0.2500	8	0.5000

Table 2 shows that our scheme has higher visual quality than other schemes at the same QF. For RDHINN (Shang et al., 2023), our scheme performs better at the same QF.

4.5 Analysis of anti-JPEG compression ability

Fig. 9 shows the ER performance under different QFs at different capacities. For the convenience

Table 2 Visual quality and ER comparison

QF	Method	Secret type	PSNR	SSIM	ER
65	RDHINN	Image	<u>33.02</u>	0.906	–
	EFDR	Image	31.47	<u>0.913</u>	–
	FNNS	Binary	28.35	0.852	0.000 067
	Ours	Binary	**38.12**	**0.969**	0.000 665
75	RDHINN	Image	<u>33.88</u>	0.929	–
	EFDR	Image	32.16	<u>0.931</u>	–
	FNNS	Binary	28.56	0.857	0.000 065
	Ours	Binary	**38.85**	**0.976**	0.000 721
85	RDHINN	Image	<u>34.77</u>	<u>0.951</u>	–
	EFDR	Image	33.64	0.927	–
	FNNS	Binary	28.80	0.865	0.000 059
	Ours	Binary	**39.21**	**0.988**	0.000 654
90	RDHINN	Image	<u>36.23</u>	<u>0.961</u>	–
	EFDR	Image	34.25	0.925	–
	FNNS	Binary	29.10	0.874	0.000 043
	Ours	Binary	**39.40**	**0.982**	0.000 512

The best results are in bold and the suboptimal results are underlined

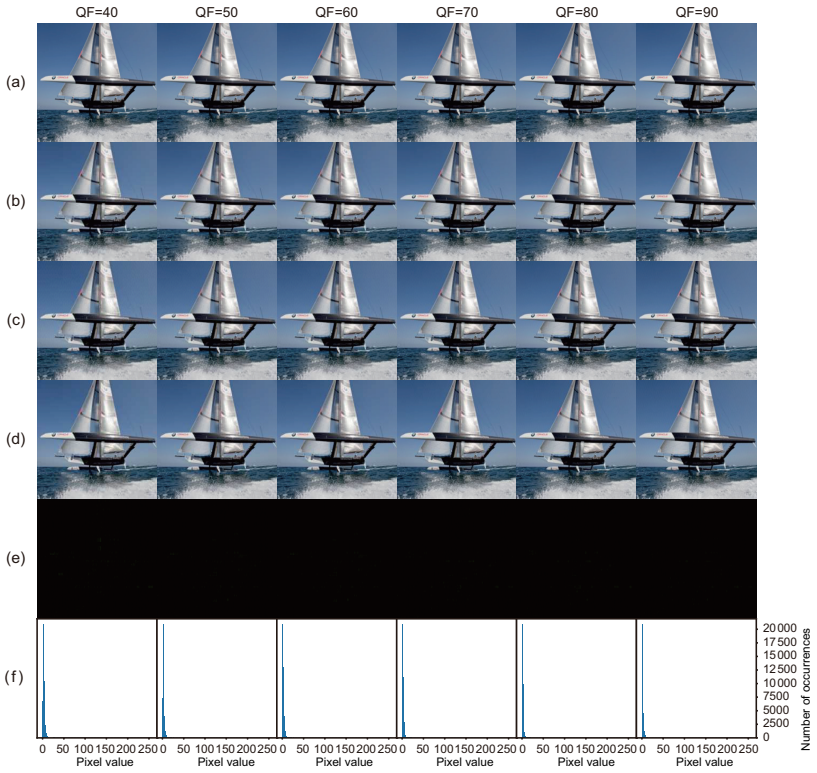

Fig. 6 The subjective visual quality under different QFs when n=3: (a) cover images; (b) quantization truncation cover images; (c) stego images; (d) compressed stego images; (e) residual of the stego images and the quantization truncation cover images; (f) histogram of the residual between the stego images and the quantization truncation cover images

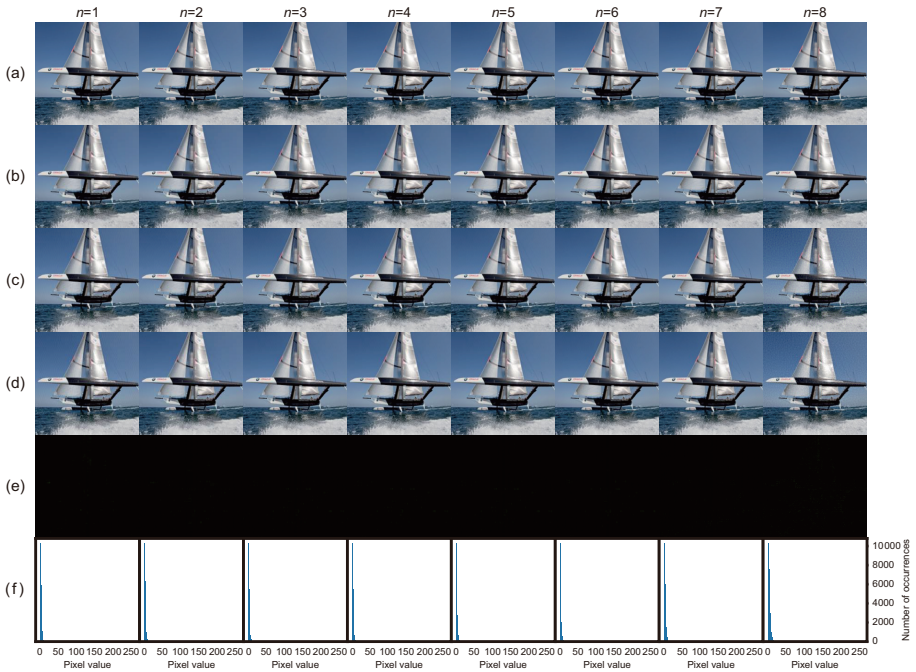

Fig. 7 The subjective visual quality under different n's when QF = 75: (a) cover images; (b) quantization truncation cover images; (c) stego images; (d) compressed stego images; (e) residual of the stego images and the quantization truncation cover images; (f) histogram of the residual between the stego images and the quantization truncation cover images

of representation, we logarithmically process the ER to clearly show its changes. It can be seen from the graph that the extraction performance of the fusion Transformer basically exceeds that of the traditional U-Net by an order of magnitude. In the case of high capacity and low QF, it can still maintain an ER close to 0. Table 3 shows that the traditional syndrome trellis code (STC)-based methods J-UNIWARD and UED can indeed recover messages under different JPEG compressions, but their ERs are high. The proposed JPEG steganography scheme can efficiently learn the inverse permutation knowledge of the compressed channel, so that the proposed scheme can learn the compression knowledge of the JPEG channel under different QF values and provide anti-JPEG compression performance.

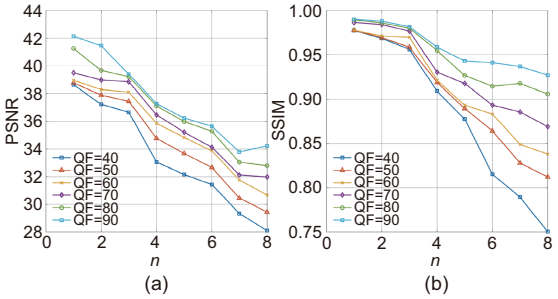

Fig. 8 PSNR (a) and SSIM (b)

4.6 Generalization performance analysis

To show the generalization performance of the model, we analyze the generalization performance of the three model variants when QF=75 and n=3. To evaluate the performance of the model on different datasets, we can use multiple different datasets to test the generalization performance of the model to ensure that the model performs well not only on the training set, but also on unseen data. If the model performs well on multiple datasets, then we can think that its generalization performance is good. After training, we use ImageNet1K, CelebA, DIV2K, and MS-coco datasets to evaluate the three model variants. We use several evaluation metrics

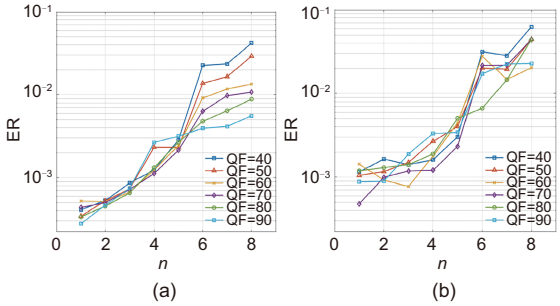

Fig. 9 Error rate (ER) under different n's using SCFformer (a) and U-Net (b)

Table 3 Comparison of the ER with the advanced anti-JPEG compression methods

QF	Method	ER			
		Payload/n=0.05/1	0.1/2	0.2/4	0.3/6
90	UED2 (Guo LJ et al., 2012)	0.3876	0.3451	0.3134	0.3003
	UED2-JCRISBE (Zhao et al., 2019)	0.1207	0.1157	0.1023	0.0889
	UED2-SRJS (Lu et al., 2021)	0.1141	0.1024	0.0979	0.0757
	J-UNIWARD (Holub et al., 2014)	0.3507	0.3314	0.3146	0.2943
	J-UNIWARD-JCRISBE (Holub et al., 2014)	0.1102	0.0842	0.0705	0.0564
	J-UNIWARD-SRJS (Holub et al., 2014)	0.0679	0.0565	0.0453	0.0392
	Ours	**0.0002**	**0.0004**	**0.0026**	**0.0039**
85	UED2	0.3786	0.3452	0.3161	0.2961
	UED2-JCRISBE	0.1196	0.1088	0.0957	0.0848
	UED2-SRJS	0.1191	0.1053	0.0897	0.0767
	J-UNIWARD	0.3561	0.3316	0.3121	0.2976
	J-UNIWARD-JCRISBE	0.1121	0.0864	0.0735	0.0601
	J-UNIWARD-SRJS	0.0635	0.0532	0.0439	0.0385
	Ours	**0.0003**	**0.0004**	**0.0013**	**0.0047**
75	UED2	0.3823	0.3668	0.3418	0.3182
	UED2-JCRISBE	0.1174	0.1057	0.0976	0.0831
	UED2-SRJS	0.1102	0.1042	0.0913	0.0784
	J-UNIWARD	0.3527	0.3372	0.3197	0.3013
	J-UNIWARD-JCRISBE	0.1089	0.0811	0.0651	0.0519
	J-UNIWARD-SRJS	0.0615	0.0517	0.0439	0.0391
	Ours	**0.0004**	**0.0005**	**0.0011**	**0.0063**

The capacity of our method is denoted by n, and the capacity of other methods is denoted by Payload. The best results are in bold and the suboptimal results are underlined

to evaluate the generalization performance of the model, including MSE, ER, PSNR, and SSIM. According to the experimental results in Table 4, our model shows good generalization performance on different datasets. The maximum value of PSNR reaches 46.06. At the same time, the average value is about 38, and even up to 42.91. Although the performance of variation B is not as good as that of variation A, it is still within an acceptable range due to the lack of DCT quantization truncation mechanism. The performance of variation C is extremely poor because it attempts to restore irreversible image degradation; however, it is still within an acceptable range.

4.7 Security analysis

To verify the security performance of the algorithm, we use StegExpose (Boehm, 2014), YeNet (Ye J et al., 2017), XuNet (Xu et al., 2016), and SRNet (Boroumand et al., 2019) steganalysis algorithms to evaluate the security performance of each quality factor using the original cover image and stego image or stego image after the compressed channel. First, we calculate the average ER and P_e value of our method and compare it with that of advanced models, as shown in Table 5. In addition, we compare area under curve (AUC) of our proposed model with that of state-of-the-art methods, as shown in Fig. 10. The results show that our method has good safety performance compared with advanced models.

Since the algorithms used by S-UNIWARD and J-UNIWARD are basically the same and their performances are basically the same, repeated comparisons will not be made.

4.8 Ablation experiment

From the ablation experimental results in Table 6, it can be seen that the DCT quantization

Table 4 Generalization performance results in four datasets

Dataset	Type	MSE	ER	PSNR		SSIM	
				Max	Average	Max	Average
ImageNet1K	Variation A	0.000 180	0.000 721	45.77	38.85	0.990	0.976
	Variation B	0.000 195	0.000 972	43.29	38.12	0.986	0.967
	Variation C	0.000 572	0.001 216	41.33	33.61	0.978	0.928
CelebA	Variation A	**0.000 064**	**0.000 139**	**46.06**	**42.91**	0.990	**0.980**
	Variation B	0.000 090	0.000 140	43.68	41.12	0.983	0.968
	Variation C	0.000 198	0.000 420	42.33	37.98	0.974	0.954
DIV2K	Variation A	0.000 222	0.000 801	43.91	38.40	0.988	0.978
	Variation B	0.000 241	0.000 725	41.48	36.97	0.988	0.971
	Variation C	0.000 741	0.001 615	38.14	32.09	0.973	0.933
MS-coco	Variation A	0.000 172	0.000 801	45.58	39.35	**0.991**	0.977
	Variation B	0.000 194	0.000 537	42.98	38.23	0.987	0.967
	Variation C	0.000 540	0.001 154	41.23	33.64	0.931	0.977

The best results are in bold and the suboptimal results are underlined

Table 5 Security analysis performed on P_e

Method	Image type	Secret type	QF_{min}	Payload↑	ER (%)↓	$P_e \to 0.5$
HidingNet (Baluja, 2020)	Natural	Image	100	**24**	92.84	0.03
UDH (Zhang CN et al., 2020)	Natural	Image	100	**24**	79.66	0.01
HiNet (Jing et al., 2021)	Natural	Image	100	**24**	43.16	0.02
DeepMIH (Guan et al., 2023)	Natural	Image	100	**24**	50.12	0.42
IRWArt (Luo et al., 2023)	Natural	Image	100	**24**	0.36	0.45
RDHINN (Shang et al., 2023)	Natural	Image	65	8	56.77	0.43
GSN (Wei et al., 2022)	Natural	Binary	100	1	2.47	0.51
Diffusion-Stego (Kim et al., 2023)	Natural	Binary	100	1	0.18	0.42
SteganoGAN (Zhang KA et al., 2019)	Natural	Binary	100	1	0.01	0.01
FNNS (Kishore et al., 2022)	Natural	Binary	65	1	0.01	0.34
StegoFormer (Li ZZ et al., 2023)	Natural	Binary	100	0.5	≥**0**	0.46
RoSteALS (Bui et al., 2023)	Natural	Binary	80	3.00E-03	≥**0**	**0.50**
Ours	Natural	Binary	**40**	0.5	≥**0**	0.51

The upward arrow indicates that the higher the value, the better the effect, and the downward arrow indicates that the smaller the value, the better the effect. In particular, the pointing arrow indicates that the closer to the value, the better the effect. The best results are in bold and the suboptimal results are underlined

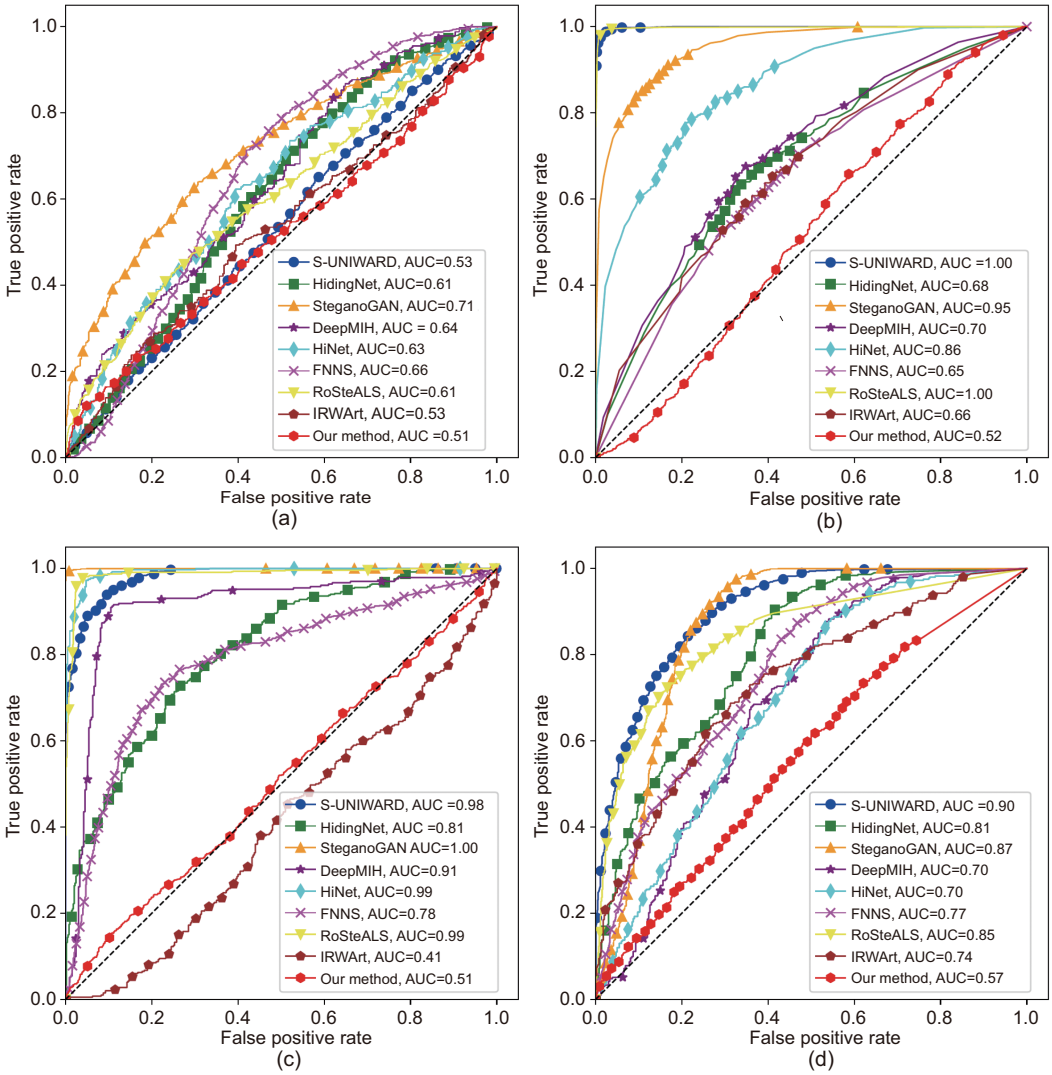

Fig. 10 The results under StegExpose (a), SRNet (b), XuNet (c), and YeNet (d) steganalysis algorithms

truncation mechanism improves the visual performance of the model and can significantly reduce the ER of message extraction. The efficient fusion Transformer can improve the security of the visual representation set of the model. In addition, the use of key control mechanism can further enhance the security of the hidden information, but the reduction of model performance and security after use is not much, within an acceptable range.

Table 6 Ablation experimental results

DCTQT	EFT	KCM	PSNR	SSIM	ER	MSE	P_e
✓	✓	✓	35.42	0.964	0.001 065	0.00 022	0.41
×	✓	✓	33.12	0.877	0.004 713	0.00 069	0.29
✓	×	✓	34.23	0.897	0.005 121	0.00 054	0.31
✓	✓	×	**38.85**	**0.976**	**0.000 721**	**0.00 018**	**0.49**
✓	×	×	35.23	0.927	0.002 126	0.00 034	0.37

DCTQT: DCT quantization truncation; EFT: efficient fusion Transformer; KCM: key control mechanism. The best results are in bold and the suboptimal results are underlined

5 Conclusions

In this work, we propose a novel anti-JPEG compression binary data hiding method based on spatial channel fusion Transformer (SCFformer), aiming to provide a large-capacity and secure data hiding method to achieve high-precision secret information recovery with low QF. At the same time, the secret information hiding technology for variable amount of secret information is realized. The experimental results show that our method can provide

high-precision secret information recovery under different QFs, and has high security, strong capacity, and high flexibility. This makes our method have broad applications and can be used in scenarios such as privacy protection and secret communication. The design of SCFformer can provide some inspiration for further research on hiding a large amount of binary data in JPEG images.

Contributors

Xintao DUAN contributed to the conceptualization. Chun LI contributed to the conceptualization, data curation, formal analysis, investigation, project administration, software development, visualization, writing, review, and editing. Bingxin WEI participated in the investigation and visualization. Guoming WU performed the validation. Chuan QIN contributed to the conceptualization, writing, review, and editing. Haewoon NAM participated in the proofreading of this paper.

Conflict of interest

All the authors declare that they have no conflict of interest.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

Abood EW, Abdullah AM, Al Sibahe MA, et al., 2022. Audio steganography with enhanced LSB method for securing encrypted text with bit cycling. *Bull Electr Eng Inform*, 11(1):185-194. https://doi.org/10.11591/eei.v11i1.3279

Baluja S, 2020. Hiding images within images. *IEEE Trans Patt Anal Mach Intell*, 42(7):1685-1697. https://doi.org/10.1109/TPAMI.2019.2901877

Bhinder P, Singh K, Jindal N, 2018. Image-adaptive watermarking using maximum likelihood decoder for medical images. *Multim Tools Appl*, 77(8):10303-10328. https://doi.org/10.1007/s11042-018-5635-z

Boehm B, 2014. StegExpose—a tool for detecting LSB steganography. https://doi.org/10.48550/arXiv.1410.6656

Boroumand M, Chen M, Fridrich J, 2019. Deep residual network for steganalysis of digital images. *IEEE Trans Inform Foren Secur*, 14(5):1181-1193. https://doi.org/10.1109/TIFS.2018.2871749

Bui T, Agarwal S, Yu N, et al., 2023. RoSteALS: robust steganography using autoencoder latent space. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition Workshops, p.933-942. https://doi.org/10.1109/CVPRW59228.2023.00100

Chen BJ, Zhou CF, Jeon B, et al., 2018. Quaternion discrete fractional random transform for color image adaptive watermarking. *Multim Tools Appl*, 77(16):20809-20837. https://doi.org/10.1007/s11042-017-5511-2

Guan ZY, Jing JP, Deng X, et al., 2023. DeepMIH: deep invertible network for multiple image hiding. *IEEE Trans Patt Anal Mach Intell*, 45(1):372-390. https://doi.org/10.1109/TPAMI.2022.3141725

Guo LJ, Ni JQ, Shi YQ, 2012. An efficient JPEG steganographic scheme using uniform embedding. Proc IEEE Int Workshop on Information Forensics and Security, p.169-174. https://doi.org/10.1109/WIFS.2012.6412644

Guo MH, Xu TX, Liu JJ, et al., 2022. Attention mechanisms in computer vision: a survey. *Comput Vis Med*, 8(3):331-368. https://doi.org/10.1007/s41095-022-0271-y

Harish NJ, Kumar BBS, Kusagur A, 2013. Hybrid robust watermarking techniques based on DWT, DCT and SVD. *Int J Adv Electr Electron Eng*, 2(5):137-143.

Hayes J, Danezis G, 2017. Generating steganographic images via adversarial training. Proc 31st Int Conf on Neural Information Processing Systems, p.1951-1960.

Holub V, Fridrich J, Denemark T, 2014. Universal distortion function for steganography in an arbitrary domain. *EURASIP J Inform Secur*, 2014:1. https://doi.org/10.1186/1687-417X-2014-1

Horé A, Ziou D, 2010. Image quality metrics: PSNR vs. SSIM. Proc 20th Int Conf on Pattern Recognition, p.2366-2369. https://doi.org/10.1109/ICPR.2010.579

Huang XH, Deng ZF, Li DD, et al., 2023. MISSFormer: an effective Transformer for 2D medical image segmentation. *IEEE Trans Med Imaging*, 42(5):1484-1494. https://doi.org/10.1109/TMI.2022.3230943

Isac B, Santhi V, 2011. A study on digital image and video watermarking schemes using neural networks. *Int J Comput Appl*, 12(9):1-6. https://doi.org/10.5120/1715-2299

Jassim FA, 2013. A novel steganography algorithm for hiding text in image using five modulus method. *Int J Comput Appl*, 72(17):39-44.

Jing JP, Deng X, Xu M, et al., 2021. HiNet: deep image hiding by invertible network. Proc IEEE/CVF Int Conf on Computer Vision, p.4713-4722. https://doi.org/10.1109/ICCV48922.2021.00469

Kim D, Shin C, Choi J, et al., 2023. Diffusion-Stego: training-free diffusion generative steganography via message projection. https://doi.org/10.48550/arXiv.2305.18726

Kishore V, Chen XY, Wang Y, et al., 2022. Fixed neural network steganography: train the images, not the network. Proc 10th Int Conf on Learning Representations, p.1.

Li S, Zhang XP, 2019. Toward construction-based data hiding: from secrets to fingerprint images. *IEEE Trans Image Process*, 28(3):1482-1497. https://doi.org/10.1109/TIP.2018.2878290

Li ZZ, Yang XY, Shen KQ, et al., 2023. Adversarial feature hybrid framework for steganography with shifted window local loss. *Neur Netw*, 164:358-369. https://doi.org/10.1016/j.neunet.2023.05.053

Liu Z, Lin YT, Cao Y, et al., 2021. Swin Transformer: hierarchical vision Transformer using shifted windows. Proc IEEE/CVF Int Conf on Computer Vision, p.9992-10002. https://doi.org/10.1109/ICCV48922.2021.00986

Lu W, Sun W, Lu HT, 2009. Robust watermarking based on DWT and nonnegative matrix factorization. *Comput Electr Eng*, 35(1):183-188. https://doi.org/10.1016/j.compeleceng.2008.09.004

Lu W, Zhang JH, Zhao XF, et al., 2021. Secure robust JPEG steganography based on autoencoder with adaptive BCH encoding. *IEEE Trans Circ Syst Video Technol*, 31(7):2909-2922. https://doi.org/10.1109/TCSVT.2020.3027843

Luo YJ, Zhou TQ, Liu F, et al., 2023. IRWArt: levering watermarking performance for protecting high-quality artwork images. Proc ACM Web Conf, p.2340-2348. https://doi.org/10.1145/3543507.3583489

Mallika, Ubhi JS, Aggarwal AK, et al., 2022. Neural style transfer for image within images and conditional GANs for destylization. *J Vis Commun Image Represent*, 85:103483. https://doi.org/10.1016/j.jvcir.2022.103483

Mou C, Xu YM, Song JC, et al., 2023. Large-capacity and flexible video steganography via invertible neural network. Proc IEEE/CVF Conf on Computer Vision and Pattern Recognition, p.22606-22615. https://doi.org/10.1109/CVPR52729.2023.02165

Shang F, Lan YH, Yang JH, et al., 2023. Robust data hiding for JPEG images with invertible neural network. *Neur Netw*, 163:219-232. https://doi.org/10.1016/j.neunet.2023.03.037

Singh SP, Bhatnagar G, 2018. A new robust watermarking system in integer DCT domain. *J Vis Commun Image Represent*, 53:86-101. https://doi.org/10.1016/j.jvcir.2018.03.006

Singh SP, Rawat P, Agrawal S, 2012. A robust watermarking approach using DCT-DWT. *Int J Emerg Technol Adv Eng*, 2(8):300-305.

Tao JY, Li S, Zhang XP, et al., 2019. Towards robust image steganography. *IEEE Trans Circ Syst Video Technol*, 29(2):594-600. https://doi.org/10.1109/TCSVT.2018.2881118

Wei P, Li S, Zhang XP, et al., 2022. Generative steganography network. Proc 30th ACM Int Conf on Multimedia, p.1621-1629. https://doi.org/10.1145/3503161.3548217

Xu GS, Wu HZ, Shi YQ, 2016. Structural design of convolutional neural networks for steganalysis. *IEEE Signal Process Lett*, 23(5):708-712. https://doi.org/10.1109/LSP.2016.2548421

Ye J, Ni JQ, Yi Y, 2017. Deep learning hierarchical representations for image steganalysis. *IEEE Trans Inform Foren Secur*, 12(11):2545-2557. https://doi.org/10.1109/TIFS.2017.2710946

Ye YX, Shan J, Bruzzone L, et al., 2017. Robust registration of multimodal remote sensing images based on structural similarity. *IEEE Trans Geosci Remote Sens*, 55(5):2941-2958. https://doi.org/10.1109/TGRS.2017.2656380

Yu XZ, Chen KJ, Wang YF, et al., 2020. Robust adaptive steganography based on generalized dither modulation and expanded embedding domain. *Signal Process*, 168:107343. https://doi.org/10.1016/j.sigpro.2019.107343

Zhang CN, Benz P, Karjauv A, et al., 2020. UDH: universal deep hiding for steganography, watermarking, and light field messaging. Proc 34th Conf on Neural Information Processing Systems, p.10223-10234.

Zhang KA, Cuesta-Infante A, Xu L, et al., 2019. SteganoGAN: high capacity image steganography with GANs. https://doi.org/10.48550/arXiv.1901.03892

Zhang Y, Luo XY, Yang CF, et al., 2015. A JPEG-compression resistant adaptive steganography based on relative relationship between DCT coefficients. Proc 10th Int Conf on Availability, Reliability and Security, p.461-466. https://doi.org/10.1109/ARES.2015.53

Zhang Y, Luo XY, Wang JW, et al., 2021. Image robust adaptive steganography adapted to lossy channels in open social networks. *Inform Sci*, 564:306-326. https://doi.org/10.1016/j.ins.2021.02.058

Zhao ZZ, Guan QX, Zhang H, et al., 2019. Improving the robustness of adaptive steganographic algorithms based on transport channel matching. *IEEE Trans Inform Foren Secur*, 14(7):1843-1856. https://doi.org/10.1109/TIFS.2018.2885438

Zhu JR, Kaplan R, Johnson J, et al., 2018. HiDDeN: hiding data with deep networks. Proc 15th European Conf on Computer Vision, p.682-697. https://doi.org/10.1007/978-3-030-01267-0_40