Information Sciences 662 (2024) 120225

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Conditional image hiding network based on style transfer

Fenghua Zhang [a], Bingwen Feng [a,*], Zhihua Xia [a], Jian Weng [a], Wei Lu [b], Bing Chen [c]

[a] College of Cyber Security, Jinan University, Guangzhou 510632, China
[b] School of Computer Science and Engineering, Ministry of Education Key Laboratory of Machine Intelligence and Advanced Computing, Guangdong Province Key Laboratory of Information Security Technology, Sun Yat-sen University, Guangzhou 510006, China
[c] School of Cyber Security, Guangdong Polytechnic Normal University, Guangzhou, 510665, China

ARTICLE INFO

Keywords:
Coverless steganography
Carrier-generative steganography
Style transfer
Transformer
Synthesized image

ABSTRACT

Various data hiding methods have been suggested to hide secret images within stego images. However, many of them could be easily detected by steganalytic tools due to their large hidden information. In this paper, we enhance the undetectability of image hiding network by mapping latent representation conditional on secret information. We extend the idea of image generation-based steganography and propose a transformer-based image hiding network that can hide a secret image with the same size as the target image. The proposed scheme uses style transferring to help map latent representation. The hiding network of the proposed scheme consists of three modules: encoding, transfer, and synthesis modules. The encoding module extracts the latent representations from content and secret images, the transfer module stylizes the latent representation, and the synthesis module fuses the latent representations to synthesize a target image with the secret image hidden in it. A new synthesis module and corresponding extraction network are developed to enhance recovery accuracy. The proposed scheme shows high image quality on both target images and recovered secret images. Furthermore, it can resist steganalytic tools and thus provide good security.

1. Introduction

Steganography embeds data into carriers in an undetectable way, allowing secret data to be transmitted over a public channel while being visible only to the recipient. There are various carriers for steganography: images, audio, video, text, and so on. Among them images are popular on the Internet, raising the attention of image steganography. Earlier image steganographic schemes such as least significant bit embedding (LSB) [1], code mapping [2], syndrome trellis code (STC) [3–5], etc., design brilliant embedders to embed secret messages into cover images. With the rise of deep learning technology, various deep learning-based image steganographic schemes [6–8] have been suggested. The balance between steganographic capacity and undetectability is essential to all these schemes.

While steganographic schemes are usually designed for binary messages, secret information can also be in the form of videos or images. However, this can make it challenging to embed the information due to the large amount of hidden data. Baluja [6] suggested that the hidden information need not be encoded perfectly, and proposed a deep learning-based method to embed a color secret image into an image of the same size. Similar schemes were suggested in succession [9–11]. Invertible networks have

* Corresponding author.
E-mail addresses: zhangfh@stu2021.jnu.edu.cn (F. Zhang), bingwfeng@gmail.com (B. Feng), xia_zhihua@163.com (Z. Xia), cryptjweng@gmail.com (J. Weng), luwei3@mail.sysu.edu.cn (W. Lu), chenbing@gpnu.edu.cn (B. Chen).

https://doi.org/10.1016/j.ins.2024.120225
Received 19 May 2023; Received in revised form 24 December 2023; Accepted 23 January 2024
Available online 1 February 2024

been recently explored to improve the quality of stego and recovered images [10,11]. These deep learning-based approaches extend traditional steganography, and show a different way to hide with a large hidden-to-host-information ratio ($\geqslant 1:1$). However, since hiding much information, they could be easily detected by steganalytic tools such as [12] and [13].

Unlike embedding-based image steganographic schemes, coverless image steganography can resist traditional steganalysis well. Coverless image steganographic schemes can be loosely categorized into two groups: schemes based on mapping rules [14,15] and based on image generation [16,8,17]. The former construct mapping rules between secret messages and the properties of images, such as BOW (Bags-of-words) model [14], ring statistic features [15], etc. Since there is no modification on transmitted images, these schemes naturally resist steganalytic tools. However, their capacity is barely satisfactory. Unlike the above approaches, image generation-based schemes synthesize nearly natural images directly from secret messages. Li et al. [16] transformed the images containing secrets into another image domain. Wei et al. [8] directly synthesized high-quality realistic images with secret messages. Peng et al. [17] utilized the denoising diffusion probabilistic model to create stego images, leading to increased accuracy in extraction. These schemes consider synthesized images rather than natural ones as the public channel. Due to the controllability of the distribution of synthesized images, image generation-based approaches present high undetectability. However, they still usually treat binary messages, and thus are difficult to hide images. Li et al. [18] hid full-size secret images into synthesized images. However, the recovery accuracy is not satisfactory. Further, it is hard to balance the visual quality between stego images and recovered secret images.

It is known that preprocessing the cover image according to the secret can enhance the undetectability [19,20]. Inspired by this, some deep learning-based steganographic schemes try to synthesize cover images more suitable for steganography. Volkhonskiy et al. [21] used DCGAN to generate cover images and then embed secret messages into these generated images to achieve high security. Shi et al. [22] improved it and suggest SSGAN to enhance the quality of generated images. However, the quality of their stego images is still limited by the generation ability of GAN. Moreover, they are designed for binary secret messages. Nevertheless, this brings us to light that we can construct an image generation-based scheme conditional on secret images to improve the balance between capacity and undetectability, as well as the balance between the visual quality of stego and recovered secret images.

Note that many steganographic schemes hide secret messages by fusing the intermediate features with secret information in the generator [6,11,8,18]. As a result, a potential solution for constructing a conditional image generation-based scheme is to map the intermediate features to a latent space closer to the secret images. In the proposed scheme, we refer to the style transfer to map the features. An et al. [23] pointed out that style transfer based on CNN architecture suffers from content leakage, which may expose secret information. [24] analyzed the causes of content leakage, yet it fails to provide generalizable solutions. In contrast, it has been shown that transformer-based style transfer models could preserve content affinity and reduce content leakage [25–28]. Inspired by this, we explore transformer-based style transfer to construct our conditional image generation-based steganographic scheme.

In this paper, we propose an end-to-end conditional image hiding network based on transformer structure, denoted as IHST. It uses style transferring to synthesize stego images conditional on the secret images. In [27], Deng et al. have verified the unbiasedness of StyTR^2, showing that it does not suffer from content leakage. Therefore, we choose this network as our backbone. IHST consists of a hiding network and a recovery network. The hiding network takes a content image and a secret image as input, and synthesizes a stylized image with the secret image hidden in it. It contains three modules: encoding, transfer, and synthesis modules. The encoding module converts the content and the secret images into their intermediate representations. The transfer module stylizes intermediate representations conditional on secret information, within which secret information can be more naturally fused. The synthesis module is of pure transformer structure, it fuses the stylized features and the secret image features and provides the final stego image. The main contributions of our work are:

1. The proposed scheme uses style transferring to map intermediate features conditional on secret information. By mapping the features to a latent space closer to secret information, we can better balance capacity and undetectability.
2. A pure transformer structure is designed in the proposed scheme. Benefitting from transformer's capturing remote dependencies, the proposed scheme can prevent the secret image content from leaking during the style transferring meanwhile improving the quality of stego images.
3. The generated stego images are indistinguishable from normal style transferring results, which can further guarantee the undetectability of the proposed scheme.

2. Related work

2.1. Transformer

Transformer architecture processes input features based on the attention mechanism. It was first applied only to language tasks [29,30], but Dosovitskiy et al. [31] kicked off the application of transformer to images by considering images as sequences of 16×16 words. Inspired by this work, researchers have proposed vision transformers for various image tasks, including image recognition [32], image generation [33,34], style transfer [27,25].

A transformer is built from an overlay of modules based on multi-head self-attention blocks (MSA) and feed-forward networks (FFN). The input is embedded as a one-dimensional sequence of tokens, described as query (Q), key(K) and value (V). The output of MSA will be computed as:

$$\mathrm{h}_i = softmax(\frac{\mathrm{Q} \cdot \mathrm{K}^T}{\sqrt{d}}) \cdot \mathrm{V} \in \mathbb{R}^{N \times d_v}, \tag{1}$$

$$output = \mathrm{W}_o \cdot \begin{bmatrix} \mathrm{h}_0 \\ \mathrm{h}_1 \\ \cdots \\ \mathrm{h}_n \end{bmatrix} \in \mathbb{R}^{N \times d_o}, \tag{2}$$

where d means the length of tokens, and $W_o \in \mathbb{R}^{d_o \times d_v}$ is the projection matrix. The feed-forward network is a fully-connected network which can be written as:

$$FFN(x) = \mathrm{W}_2 \cdot (ReLU(\mathrm{W}_1 \cdot x + b_1)) + b_2, \tag{3}$$

where $\mathrm{W}_1 \in \mathbb{R}^{d_h \times d_x}, \mathrm{W}_2 \in \mathbb{R}^{d_{out} \times d_h}$ are the weight matrices, and $b_1 \in \mathbb{R}^{d_h}, b_2 \in \mathbb{R}^{d_{out}}$ are the bias vectors. h means the length of the hidden layer.

A convolution layer has a local receptive field. Thus, the CNN-based model is inefficient for handling remote dependencies. Meanwhile, the CNN-based model exhibits a strong bias toward feature localization and spatial invariance because of the shared convolutional filter weights at all locations. In comparison, the attention-based transformer allows interaction between inputs at different locations. Therefore, it can easily access global information about the input features and learn complex relationships between its inputs without limitations. Such capability of handling remote dependencies facilitates the adaptation of our model to hide secret images in suitable regions in steganography. Moreover, according to [33], its global receptive field makes it possible to generate locally explicit as well as globally consistent styles.

2.2. Style transfer

Style transfer aims to transfer the artistic style from a style image to a content image without changing the original semantic information in the content image. In 2015, Gatys et al. [35] suggested a neural algorithm for high perceptual quality of art style transfer. However, this work formulates each transfer process as an optimization problem, which is time-consuming. To solve this problem, Johnson et al. [36] replaced the optimization process with pre-trained feed-forward networks. Although this approach substantially improves operational efficiency, its model is limited to a particular style and requires retraining for different styles. AdaIN [37] suggested by Huang et al. first de-styled the content image and then completed the stylization by using the mean and variance of the style image features. This scheme can achieve arbitrary style transfer. CycleGAN has showcased its capability in style transfer [38], but it still suffers from being style-bound. An et al. [23] found that CNN-based models struggle with content leakage due to bottlenecks in their local receptive fields. Consequently, researchers turned their attention to transformer models with global receptive fields. Various transformer-based style transfer models [26,25,27] are suggested, which reduce content leakage and can provide flexible and effective stylization.

There are several style transfer-based steganographic schemes. STNet [39] created artistic images from style and content images while embedding secret messages into style features. ISTNet [40] improved STNet to embed a grayscale image into a color image with better security. Nonetheless, their capacity is not comparable to those schemes that can hide color images. Furthermore, the style transfer in these schemes is independent of the secret images, which means it cannot ensure that the resulting images are more suitable for steganography. Moreover, the presence of secret information degrades the performance of style transfer.

In the proposed scheme, we expect that style transfer could help develop a better steganographic scheme based on image generation by mapping the intermediate representation to a latent space more suitable for embedding secret images. Furthermore, the proposed scheme should support arbitrary style transfer, and the quality of synthesized images should be high. StyTR^2 [27] can ensure high-quality synthesized images. StyTR^2 [27] satisfies these requirements and eliminates content leakage. Therefore, we choose it to develop our steganographic scheme.

3. Proposed approach

3.1. Conditional intermediate representation

The proposed scheme tries to generate a stego image directly from a secret image of the same size. Given a secret image $I_s \sim P(\mathbf{I}_s)$, it outputs a stego image $I_y \sim G(z, I_s)$ where $G()$ is the generator and z is the other necessary input besides I_s. The statistic security of steganography necessitates that the distribution of stego images should be close to that of innocent synthesized images $I_x \sim Q(z)$ [8]. That is, $G(z, I_s)$ should optimize

$$\min_G \left(\mathbb{E}_{I_s \sim P(\mathbf{I}_s)} [\log(1 - D(G(z, I_s)))] \right) \tag{4}$$

where $D()$ is certain steganalytic scheme that can assign label 1 to $I_x \sim Q(z)$ and 0 to $I_y \sim G(z, I_s)$. From the perspective of information theory, the performance of $D()$ is limited by the KL divergence $\mathcal{D}_{KL}(Q(z) \| G(z, I_s))$.

Usually, the generator of an image-generation-based scheme needs to create intermediate features f before it synthesizes the final stego image, that is, $G = G_1 \circ G_2, f \sim G_1(z, I_s), I_y \sim G_2(f, I_s)$. If the intermediate features can be created conditional on the secret, denoted as $f \sim G_1(z, I_s | I_s)$, it can be derived that

$$\mathcal{D}_{KL}(Q(z) \| G(z, I_s))$$

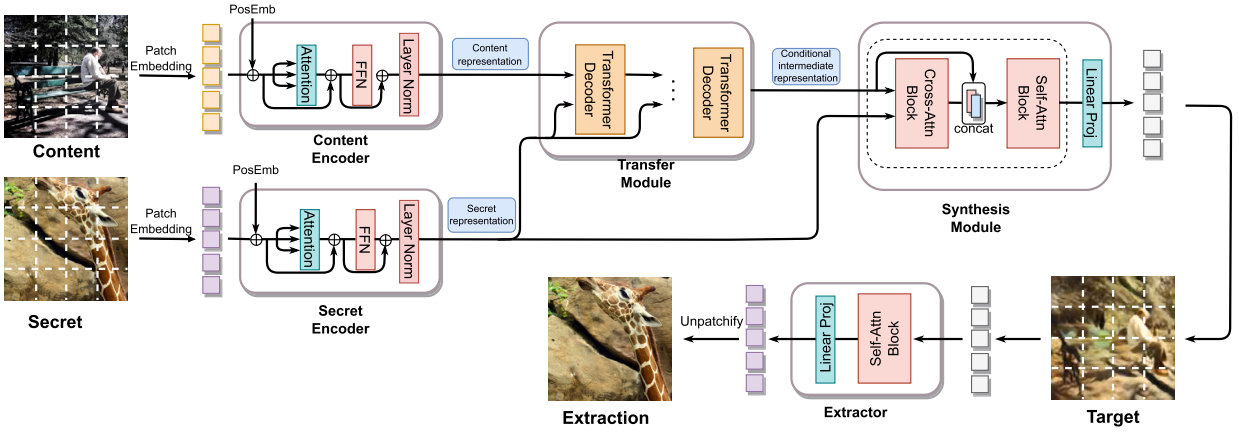

Fig. 1. The framework of IHST.

$$= \mathbb{E}_{I_x \sim Q(z)} \log p(I_x) - \mathbb{E}_{I_x \sim Q(z)} \mathbb{E}_{I_y \sim G(z, I_s)} \log p(I_y) \tag{5}$$

$$\begin{aligned} = & \mathbb{E}_{I_x \sim Q(z)} \log p(I_x) \\ & - \mathbb{E}_{I_x \sim Q(z)} \mathbb{E}_{I_y \sim G_2(f, I_s)} \mathbb{E}_{f \sim G_1(z, I_s | I_s)} \log p(I_y) \\ & + \mathbb{E}_{I_x \sim Q(z)} \mathbb{E}_{I_y \sim G_2(f, I_s)} \mathbb{E}_{f \sim G_1(z, I_s | I_s)} \log p(I_y) \\ & - \mathbb{E}_{I_x \sim Q(z)} \mathbb{E}_{I_y \sim G_2(f, I_s)} \mathbb{E}_{f \sim G_1(z, I_s)} \log p(I_y) \end{aligned} \tag{6}$$

$$= \mathcal{D}_{KL}(Q(z) \| G_1(z, I_s | I_s) \circ G_2(f, I_s)) + \mathcal{I}(G(z, I_s); P(\mathbf{I}_s)) \tag{7}$$

$$\geq \mathcal{D}_{KL}(Q(z) \| G_1(z, I_s | I_s) \circ G_2(f, I_s)) \tag{8}$$

where Eq. (8) satisfies because mutual information $\mathcal{I}(G(z, I_s); P(\mathbf{I}_s)) \geq 0$. It indicates that stego images generated from these intermediate features are more secure. On the other hand, it can be found that

$$\begin{aligned} & \mathcal{I}(P(\mathbf{I}_s); G(z, I_s)) \\ = & \mathcal{I}(P(\mathbf{I}_s); G_1(z, I_s) \circ G_2(f, I_s)) \end{aligned} \tag{9}$$

$$\begin{aligned} = & \mathcal{I}(P(\mathbf{I}_s); G_1(z, I_s | I_s) \circ G_2(f, I_s)) \\ & - \mathbb{E}_{I_y \sim G_2(f, I_s)} \mathbb{E}_{f \sim G_1(z, I_s | \sim I_s)} \mathbb{E}_{I'_s \sim p(I_s | I_y)} \log p(I'_s | I_y) \end{aligned} \tag{10}$$

$$\leq \mathcal{I}(P(\mathbf{I}_s); G_1(z, I_s | I_s) \circ G_2(f, I_s)) \tag{11}$$

It means that the obtained conditional intermediate features can also increase the relation between the secret and the stego images, which could improve the recovery accuracy of the secret. In fact, preprocessing the cover according to the secret has been widely used in steganographic schemes when the embedded secret messages are binary [19]. However, when the secret messages are meaningful images, it is difficult to preprocess the cover to fit the secret. Nevertheless, in image-generation-based schemes, the conditional intermediate features can be obtained by style transferring the intermediate features with the style of secret images.

The proposed Image Hiding network with Style Transfer (IHST) is an end-to-end trainable framework. There are two networks in IHST, the hiding network and the recovery network, as shown in Fig. 1. The hiding network takes a content image and a secret image as input. It aims to generate an artistic style image that resembles the content image while concealing the secret image. The recovery network recovers the secret image from the stego image to complete secure communication. They are detailed in the following two subsections.

3.2. Hiding network

Hiding network $G(I_c, I_s)$ uses content image I_c and secret image I_s to synthesize a target image I_y with I_s hidden in it. It consists of encoding, transfer, and synthesis modules. Encoding module converts the content and the secret images into the intermediate representations. Transfer module generates a stylized representation by rendering the content image features using the style of the secret image. Synthesis module fuses the conditional intermediate representation and the secret image features and outputs a target image with the secret image hidden in it. These submodules are detailed in the following.

Encoding Module. This module extracts semantic and style information from the input I_c and I_s, respectively, to facilitate the subsequent style transfer and image synthesis. The secret image I_s serves as both the source input of style transfer and the secret information to be hidden. Transformer encoders have been widely used due to their robust feature extraction capability and global receptive field [31,41,42]. As a result, we use them to construct the encoding module.

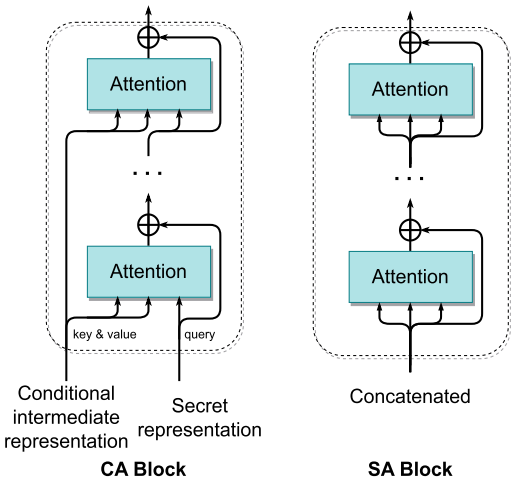

Fig. 2. The architecture of CA Block and SA Block.

The employed transformer encoder is composed of a patch embedding, a positional embedding, and a stack of sequentially connected multi-head self-attention (MSA) and feed-forward neural network (FFN) with a layer normalization (LN). Given an input image $I_\epsilon \in \mathbb{R}^{n_H \times n_W \times 3}$, where $\epsilon \in \{c, s\}$ and (n_H, n_W) is the resolution of the image, it is encoded to an intermediate representation z_ϵ.

Transfer Module. This module receives the intermediate representations output by the Encoding Module, and transfers the content intermediate representation z_c conditional on the secret representation z_s. The output of this module, f, is a conditional intermediate representation containing the target style and the semantic of the content image.

We choose the transformer decoder of StyTR^2 [27] as the backbone of the transfer module, as it does not suffer from content leakage and supports arbitrary style transfer. It can efficiently perform the image style transfer task that migrates the input z_c to the domain of z_s. The process of transfer module (T) can be expressed as follows.

$$f = T(z_c, z_s). \tag{12}$$

Synthesis Module. This module generates target images indistinguishable from ordinary stylized images. It receives the conditional intermediate representation output by Transfer Module, and decodes it into an image meanwhile concealing the secret image.

TransGAN [34] proposed by Jiang et al. has demonstrated that the encoder of the transformer can not only extract features, but also generate images from latent features. Therefore we synthesize images using attention blocks with similar structures. Note that most visual transformers divide images into patches with positional embedding to alleviate computation costs [34,31]. Since our synthesis module takes reference to the features of two images, each patch should be fused according to information from the global image rather than from some paired patches. Consequently, we introduce cross-attention to captain relations across patches.

The designed synthesis module consists of two parts: an attention part and a linear projection part (FC), as shown in Fig. 1. The attention part is divided into two blocks, the cross-attention block (CA) and the self-attention block (SA). In Section 4.5 we have tested different architectures for Synthesis module, and shown that the selected architecture can provide the best performance.

Both CA and SA consist of multiple attention layers, as illustrated in Fig. 2. CA is built up of several attention layers with the same structure. The key and value of each layer are the conditional intermediate representation f, while the query is from the previous layer's output. The query of the first layer is the secret representation z_s. The output of CA will be concatenated with f and subsequently fed into SA. Following the decoding process within the self-attention block, the resulting representation is unflattened into image blocks by FC. These blocks are combined to form the target stego image I_y, as expressed in Eq. (13)

$$I_y = FC(SA(f \oplus CA(f, z_s))). \tag{13}$$

The conditional intermediate representation f contains the content information of the content image and the style information of the secret image, which thus has a potential correspondence with the secret image. By computing the similarity between f and z_s through the CA block, we project f into the secret space. It makes it easier for SA to further fuse f and z_s. Section 4.5 will experimentally verify that the connection of CA and SA will provide good performance.

3.3. Recovery network

A recovery network $E(I_y)$ is customized to recover the concealed secret image from I_y. It consists of multiple transformer encoder layers and a linear projection layer stacked on top of each other. The recovery network operates in two phases. Initially, it extracts

the latent feature representation of the secret image from the target image. Subsequently, it decodes the feature map into an image. It can be expressed as

$$\tilde{I}_s = E(I_y), \tag{14}$$

where $\tilde{I}_s$ is the recovered image.

3.4. Loss function

There are two tasks for the proposed network. First, the target image I_y should be difficult to distinguish from normal artistic images. Second, the recovered image I_{ext} should be as close as possible to the secret image I_s. As a result, the total loss function should balance four losses, formed as

$$Loss = \lambda_1 L_{st} + \lambda_2 L_{id} + \lambda_3 L_{sim} + \lambda_4 L_{rec}, \tag{15}$$

where L_{st}, L_{id}, L_{sim}, and L_{rec} denote the style transfer loss, identity loss, similarity loss, and recovery loss, respectively. $\lambda_1 \sim \lambda_4$ are the weights of the corresponding loss.

Identity Loss and Style Transfer Loss. We employ the two losses defined in [27] to ensure that the generated images are indistinguishable from normal artistic images. The style transfer loss is originally defined on the target images. However, we find that this makes the proposed scheme highly unstable. To deal with this problem, we apply the style transfer loss to the conditional intermediate representation f produced by Transfer Module. Specifically, f is first decoded by a CNN-based decoder network, DE, defined in [27], and then input into the loss function, formed as

$$L_{st} = L'_{st}(DE(f), I_s), \tag{16}$$

where L'_{st} is the style transfer loss function defined in [27].

Similarity Loss. This loss ensures that the synthesized target images avoid deviating from the domain of the artistic style and exposing the secret images. We constrain the synthesized images according to the styled representation f and define the similarity loss as

$$L_{sim} = \|DE(f) - I_y\|_2. \tag{17}$$

Recovery Loss. This loss constrains the model to recover the secret image accurately. It is simply formed as an L2 distance between recovered image $\tilde{I}_s$ and secret image I_s.

$$L_{rec} = \|\tilde{I}_s - I_s\|_2. \tag{18}$$

4. Experimental result

4.1. Implementation details

The proposed scheme is trained on Pytorch 1.11.0 and CUDA 11.3 with two Nvidia 3080Ti GPU. It is evaluated on COCO [43] and WikiArt [44] datasets. All the images have been resized to 256×256 and randomly cropped to 128×128. The patch size of patch embedding is set to 8. A standard Adam with an initial learning rate 5×10^{-4} is used as the optimizer. We set $\lambda_1 = 7, \lambda_2 = 1, \lambda_3 = \lambda_4 = 80$ in Eq. (15). The number of training iterations is $160K$. The size of the mini batch is set to 8.

4.2. Naturalness of generated target images

Our scheme directly generates target images. Furthermore, it uses style transferring to handle capacity and undetectability. Given this, we evaluate the naturalness of our scheme from two aspects: image generation and style transfer. The former asks the generated image to be natural, while the latter requires the scheme to act like other style transfer methods.

We compare the proposed scheme with various SOTA image steganographic schemes using different strategies, including embedding-based steganographic schemes (UDH [9]), coverless steganographic schemes (ECIH [16] and HCCS [18]), and carrier-generating-based steganographic schemes (ISTNet [40]).

4.2.1. Similarity to natural and artistic images

The similarity to natural images is herein measured with Fréchet inception distance (FID) [45], which has been widely used as the metric of the distance between two image domains [8,46]. We still employ the COCO dataset as the natural image dataset and compare the distances from the stego images generated by different schemes to it. In all these schemes, both cover/content images and secret images are randomly selected from the COCO dataset to ensure the naturalness of generated images. Table 1 lists the comparison results. It can be observed that the scores obtained by our scheme are better than those of the compared schemes except UDH. UDH embeds secret images by marginally altering cover images and thus can achieve the best FID scores. Nevertheless, our scheme exhibits naturalness similar to other image-generation-based steganographic schemes.

Table 1
FID between target images and natural images.

	Proposed	ECIH [16]	HCCS [18]	ISTNet [40]	UDH [9]
FID	68.81	375.94	93.19	90.36	31.72

Table 2
FID between target images and artistic images.

	Proposed	StyTR^2 [27]	ArtFlow [23]	AdaIN [37]
FID	67.71	74.38	92.85	27.98

Table 3
Quantitative comparison in terms of content loss and style loss.

	Proposed	StyTR^2 [27]	ArtFlow [23]	AdaIN [37]
Content loss	1.5675	2.0072	0.9610	1.9785
Style loss	5.7617	5.3841	11.5669	6.1979

Table 4
Averaged PSNR and SSIM of recovered images from different schemes.

	Proposed	ISTNet [40]	UDH [9]	ECIH [16]	HCCS [18]
PSNR	**35.9**	30.5	34.3	20.9	26.3
SSIM	0.96	0.75	**0.97**	0.94	0.91

On the other hand, style transfer applications usually generate artistic images. Therefore, we still need to evaluate the similarity of generated images to this type of images. We compare our method to three SOTA style transfer methods, that is, StyTR^2 [27], ArtFlow [23], and AdaIN [37]. Herein, the cover/content images are randomly selected from the COCO dataset, while the secret/style images are randomly selected from the WikiArt dataset. The overall results in the term of FID are shown in Table 2. It shows that our method can obtain FID scores competitive with these methods. Strangely, AdaIN obtains the lowest scores, possibly because its normalization is better suited for the features used in FID metric. In general, our approach can generate images that are natural in both the context of natural images and artistic images.

4.2.2. Similarity to style transferred images

This section evaluates the style transfer performance of the proposed scheme by comparing it with StyTR^2, ArtFlow, and AdaIN. The image sizes of these schemes have been adjusted to match those in our scheme.

Fig. 3 compares the synthesized images obtained by different schemes. It shows that the synthesized images of each algorithm are slightly different in detail. Although our scheme has embedded secret images of equal size, the quality of synthesized images is still competitive. Our effort on steganography will leave the synthesized image lacking in detail. Notwithstanding, it is difficult to distinguish our results from those obtained by other schemes.

Observing that non-artistic images are more applicable for steganography than artistic images, we also compare the performance when the style images are not artistic, as shown in the last three rows of Fig. 3. It can be seen that the proposed scheme can still accurately capture the style in these images and present stable performance.

We then quantitatively evaluate our scheme on style transfer. The content and style loss are employed as metrics following the methodology described in [27,23]. Table 3 lists the average content and style loss for each scheme. There is no significant difference among the compared schemes except that ArtFlow does not perform very well in style loss. Our IHST generates images with medium levels of both content loss and style loss, making it a good scheme for style transfer tasks.

4.3. Recovery accuracy

The accuracy of secret image recovery is then evaluated with metrics Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM). We compare it with UDH, ECIH, HCCS, and ISTNet. Both cover and secret images are randomly selected from the COCO dataset. Table 4 lists the PSNR and SSIM scores averaged over 1000 recovered images. It can be seen that our IHST outperforms coverless steganographic schemes and carrier-generating-based schemes. However, its SSIM scores are slightly worse than UDH. We further compare the visual quality of recovered images obtained by our IHST and UDH. As shown in Fig. 4, it can be observed that there is a slight discontinuity in the smooth region in some images recovered by the IHST. Nevertheless, IHST can provide recovery accuracy comparable to UDH in most cases.

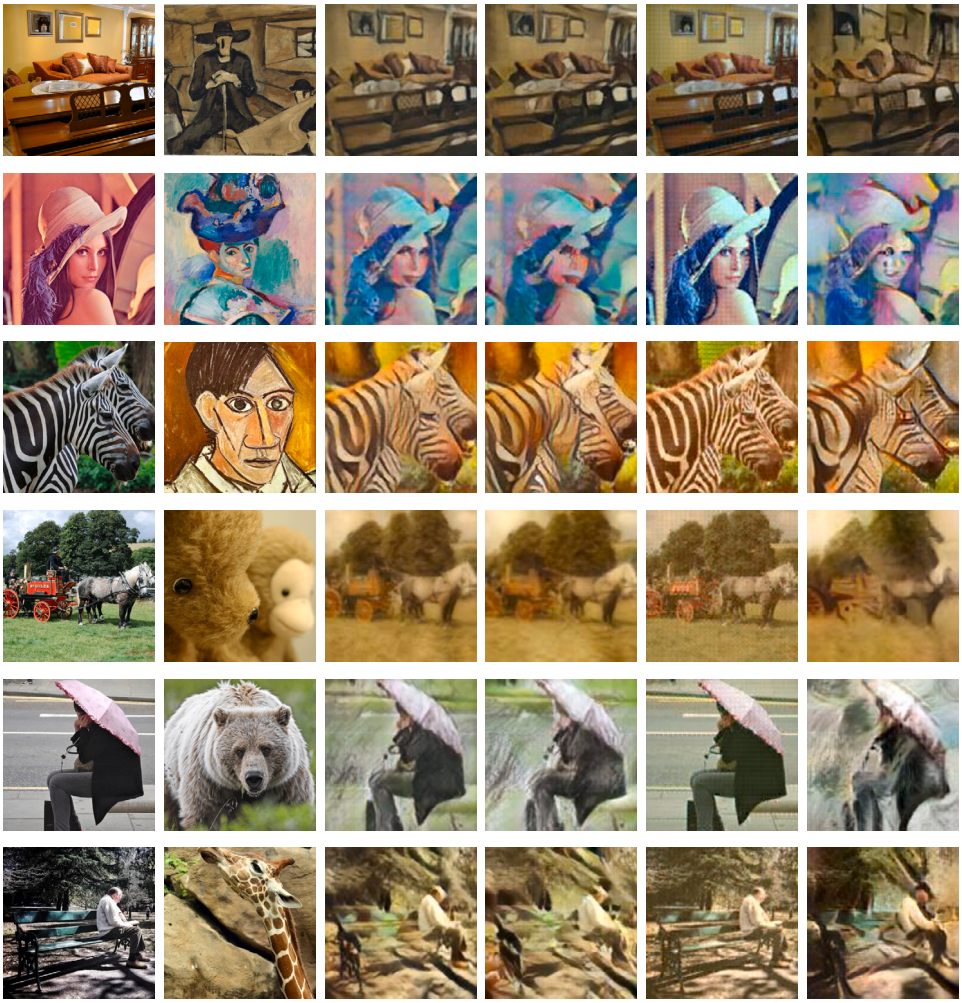

Fig. 3. Visual comparison of style transfer results. Images in a) are the content images, in b) are the style images, in c), d), e), and f) are the synthesized images obtained by the proposed IHST, StyTR^2, ArtFlow, and AdaIN, respectively. The content images are from COCO dataset. The style images in the first three rows are from WikiArt dataset, while those in the last three rows are from COCO dataset.

4.4. Security

We verify the security of our scheme in three ways: secret leakage resistance, perceptual similarity to cover images, and statistical undetectability.

4.4.1. Secret leakage resistance

The first concern on IHST is whether style transfer would leak the content of secret images. We use the detection method of content leakage described in [27] to verify the security of style transfer. Five rounds of embedding with the same secret image are performed on our scheme, where each round is carried out on the output of the previous round. The synthesized images obtained in different rounds are demonstrated in Fig. 5. To better evaluate the proposed scheme, we further demonstrate the results obtained by AdaIN. It can be observed that, with the increment of stylizing iterations, the images generated by AdaIN are progressively difficult to understand. Their content is gradually replaced by stylistic patterns. This may be due to its biased decoding training, as stated in [24]. On the other hand, IHST largely preserves the original content. Although multiple rounds of embedding produce checkerboard patterns in synthesized images, the secret content, such as item outline and spatial structure, is not revealed. In IHST, we use style transfer based on StyTR^2, whose content leakage was confirmed in [27]. Therefore, our IHST does not expose the content of secret images.

In addition to content leakage detection, we also use ManTra-Net [47] to detect possible traces of secret image embedding in the synthesized images. ManTra-Net is a deep learning-based pixel-level image forgery detection network that exports the probability

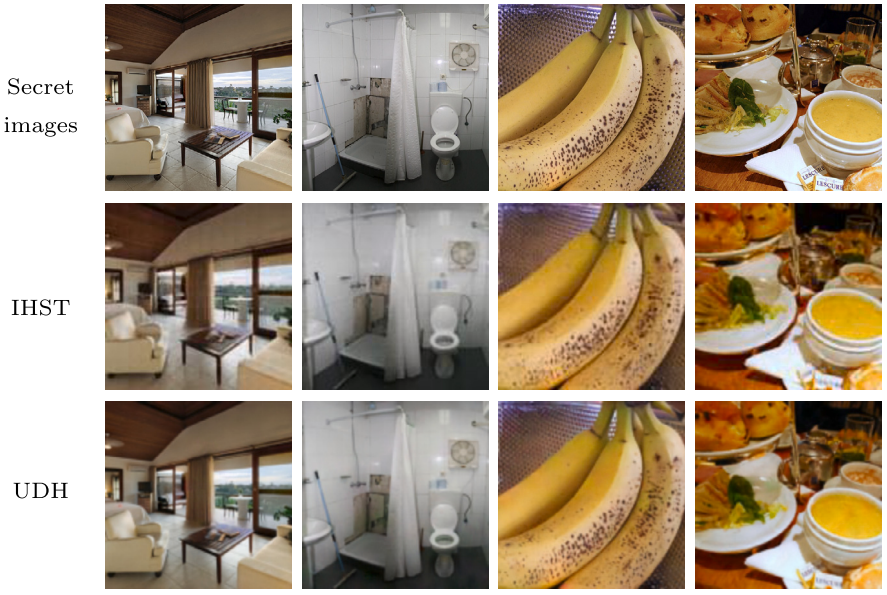

Fig. 4. Visual comparison of secret images recovered by IHST and UDH.

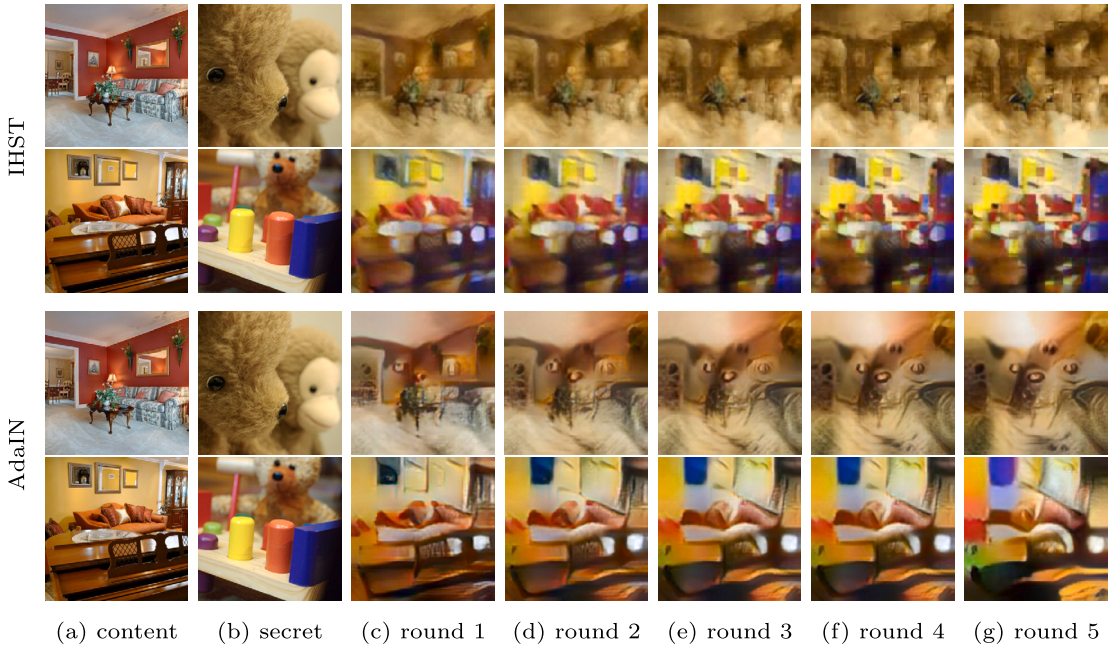

Fig. 5. Visual detection by multiple rounds of message embedding. Images in a) are the content images, in b) are the secret images, in c)-g) are the synthesized images obtained in round 1-5, respectively.

of suspicion on each pixel. It has demonstrated that this network can also effectively detect image steganography [10,18]: some embedding-based steganographic schemes such as [6] will be easily detected by ManTra-Net. We compare the detection results on the images generated by IHST and common stylized images. The comparison results shown in Fig. 6 illustrate that the suspicious regions detected in our generated images are distributed similarly to the common stylized images. It indicates that the secret content is well concealed under the cover of stylization.

4.4.2. Perceptual similarity to cover images

Steganographic schemes should output stego/target images perceptually indistinguishable from cover images to arouse suspicion. Therefore, this section evaluates the image quality of target images. Considering the publicly transmitted images are innocent stylized

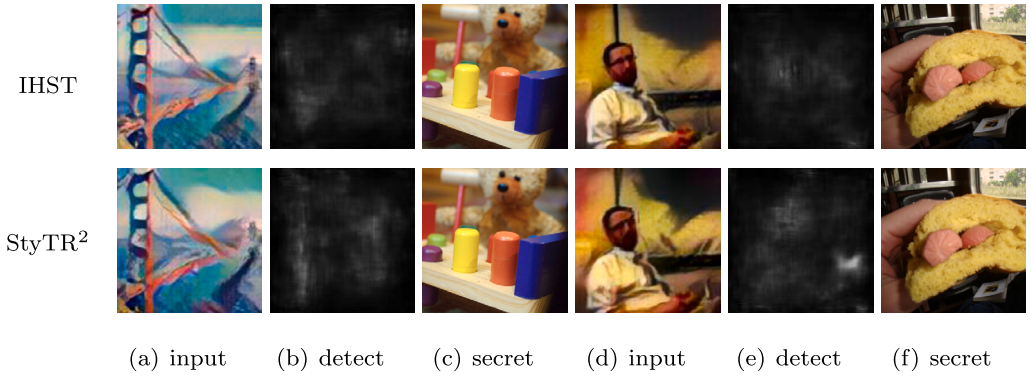

Fig. 6. Comparison of visual detection by ManTra-Net.

Table 5
FID between target images and cover images as well as stylized images.

	Cover images	Styled images
FID	7.93	13.73

images in style transfer applications, we take the stylized images without any hidden messages as the "cover images" following the idea in [8,46]. It is carried out by retraining our network to generate stylized results without hiding secret images.

Furthermore, we perform the same style transferring on the same content images in the backbone network [27], which gives innocent stylized images. Then the perceptual similarity between target images and these cover images and innocent styled images are evaluated by using the FID metric. The experimental results are reported in Table 5. It shows that the target images exhibit remarkable similarity with the cover images and innocent styled images. As a result, it is hard to detect whether secret data exists in the generated images.

4.4.3. Statistical undetectability

Practical attackers may construct a content image dataset and train a steganalytic tool to detect whether the stylized images contain secret information if he has the knowledge of content images used for image hiding. In view of this, we evaluate the statistical undetectability of our IHST on several advanced steganalytic tools. UDH, HCCS, and ECIH are used for comparison. Furthermore, in order to evaluate the effectiveness of image generation-based steganographic schemes on hiding images compared with traditional embedding-based schemes. We use multi-layer STC [3] to mimic image hiding. It is performed by first lossy compressing a secret image using JPEG to obtain a compressed image with a similar PSNR score as the recovered image obtained by the proposed scheme, then embedding the bitstream of the compressed image into a cover image by multi-layer STC. Experimentally we find that, in general, a 3-layer STC is needed to finish the mimicking.

We first compare these schemes using StegExpose [48], an open-source steganalysis toolkit containing powerful methods like RS analysis, Chi Square Attack, etc. It is a benchmark for anti-steganalysis evaluation in many previous researches [18,10]. A total of 500 cover images and 500 stego images are generated/collected using the COCO dataset for each scheme. The ROC curves of each scheme are displayed in Fig. 7. The closer the curve is to the optimal ROC curve (the counter diagonal in the figure), the better the undetectability of the scheme. It can be observed that the ROC curve of IHST is the closest to the optimal ROC curve, indicating that our scheme provides the best undetectability against StegExpose.

In recent years, deep steganalytic methods have been developed and presented significantly better performance compared with traditional steganalytic methods. Therefore, we use two advanced deep steganalytic networks, SRNet [12] and XuNet [13], to test the undetectability of these steganographic schemes. Weng et al. [49] suggest that the number of stego/cover samples required for training steganalytic tools can be considered as an indicator of the security of a steganographic scheme. Following their methods, we retrain both SRNet and XuNet with various numbers of training samples, and observe the changing in detection accuracy as the number of training images increases. Both the training and the testing images were randomly collected each time. The detection accuracies after training are shown in Fig. 8. It can be observed that IHST has the slowest increase in detection accuracy. Other methods in SRNet achieved a detection accuracy of 90% with less training data. The case in XuNet is similar. It suggests that more training samples are required to accurately detect the proposed scheme. As a result, our IHST presents high undetectability even if the used content images are leaked. However, we find that, if the same images are used in both training and testing in the experiments, the detection accuracy will rise dramatically, which performs similar to ML-STC. It indicates that the security will degrade if an attacker has exact knowledge of the content image used for steganography. This is coincident with the conclusion in traditional steganography.

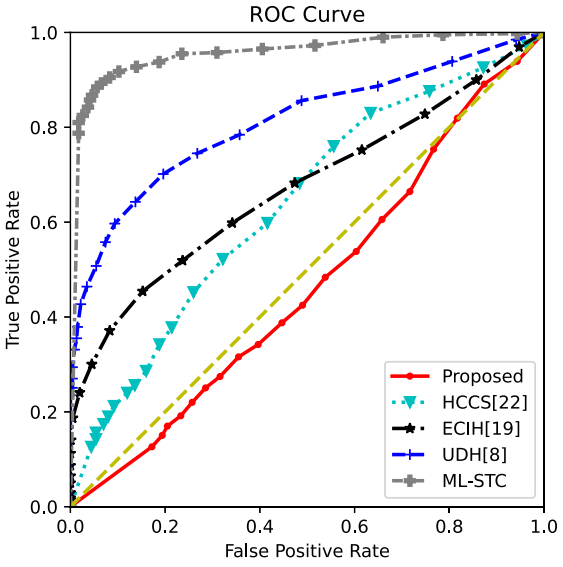

Fig. 7. ROC curves of different steganographic schemes.

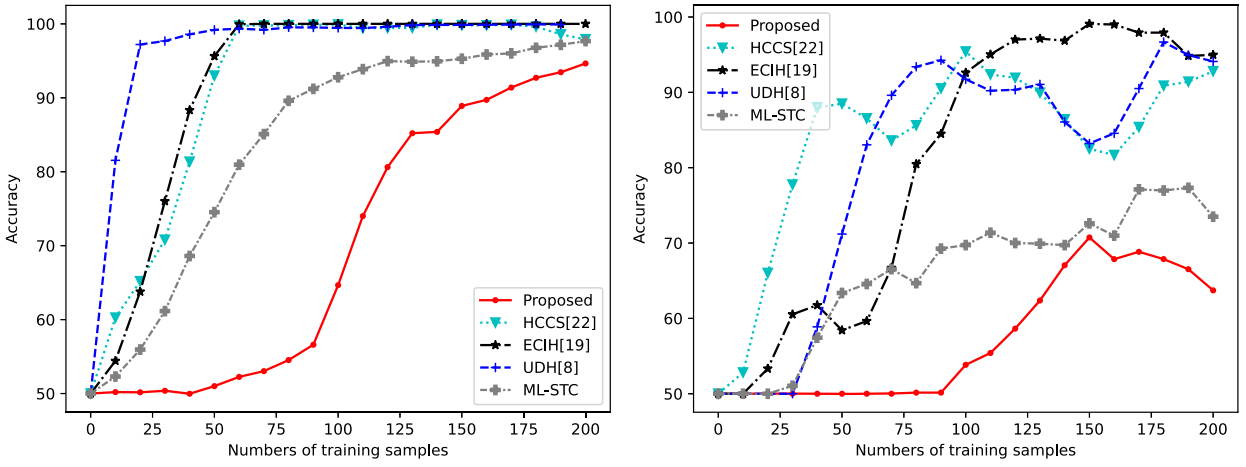

Fig. 8. Detection accuracy by using a) SRNet and b) XuNet during training process.

4.5. Ablation experiments

4.5.1. Effectiveness of transfer module

We evaluate the effectiveness of style transfer in the proposed scheme. A transformer-based image hiding network is constructed here by removing the style transfer module in IHST. It contains only encoding and synthesis modules to embed secret images directly into cover images. We employ two types of cover images: nature images directly collected from the COCO dataset (denoted as IH), and stylized images using StyTR^2 on the COCO dataset (This style transfer does not consider the secret images, denoted as IH2). These two embedding strategies are compared with IHST with respect to recovery accuracy as well as security. The recovered images from these methods are compared in Fig. 9, where we also report the PSNR and SSIM scores. As can be seen in Fig. 9, IHST brings a significant improvement in recovery accuracy.

We then compare the undetectability by using SRNet. The cover and stego images generated by IH are trained on SRNet. All the experiment setups are the same as those when detecting IHST. The detection accuracy in each training is depicted in Fig. 10. It illustrates that introducing style transfer according to the secret can enhance undetectability. This confirms the analysis in Section 3. Notably, the proposed scheme achieves considerable undetectability without using any steganalyzer or discriminator for adversarial training.

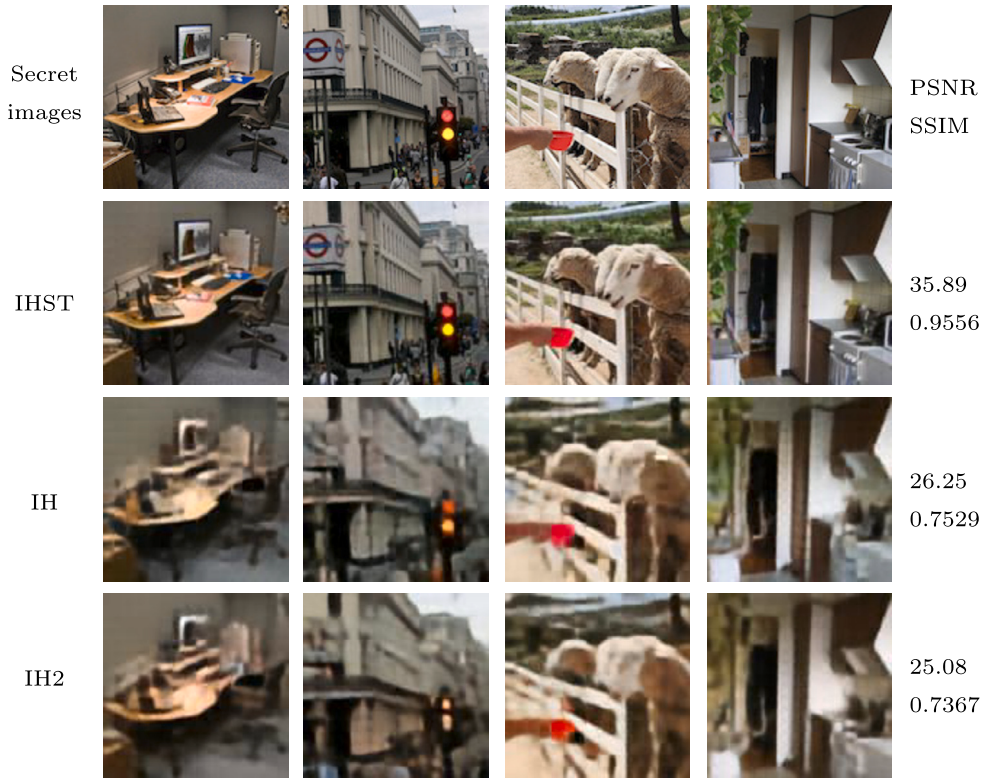

Fig. 9. Comparison of recovered images by IHST, IH, and IH2.

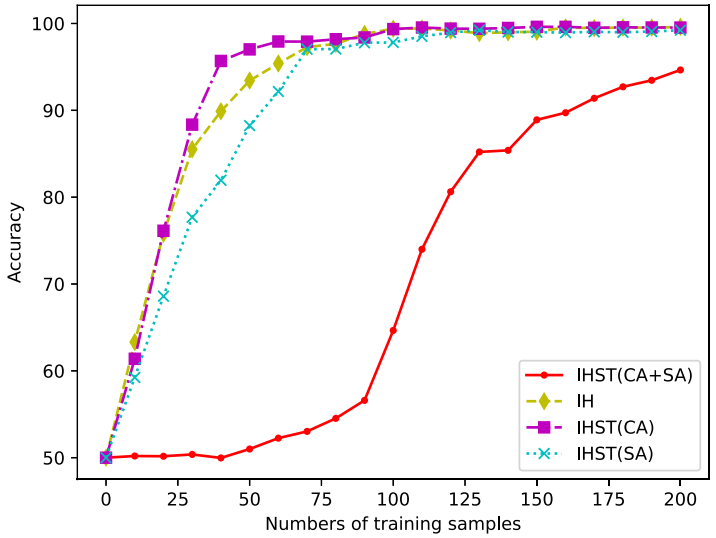

Fig. 10. Detection accuracy of IH, IHST(CA), IHST(SA), and IHST(CA+SA) during training SRNet.

4.5.2. Performance of different architectures for synthesis module

We test several different transformer architectures to explore the best structure for the synthesis module. Three choices can be used to construct the synthesis module, as shown in Fig. 11. Fig. 11(a) shows the module with cross-attention layers only. Fig. 11(b) shows the module with self-attention layers only. Fig. 11(c) shows the cross-attention layer connected in series with the self-attention layer. To ensure fairness, we set the total layer number to be the same for all three architectures. Three new IHST models are rebuilt with these architectures, denoted as IHST(CA), IHST(SA), and IHST(CA+SA), and trained in the same experiment setting. Their performances on the quality of target and recovered images, as well as undetectability are then tested using the COCO dataset.

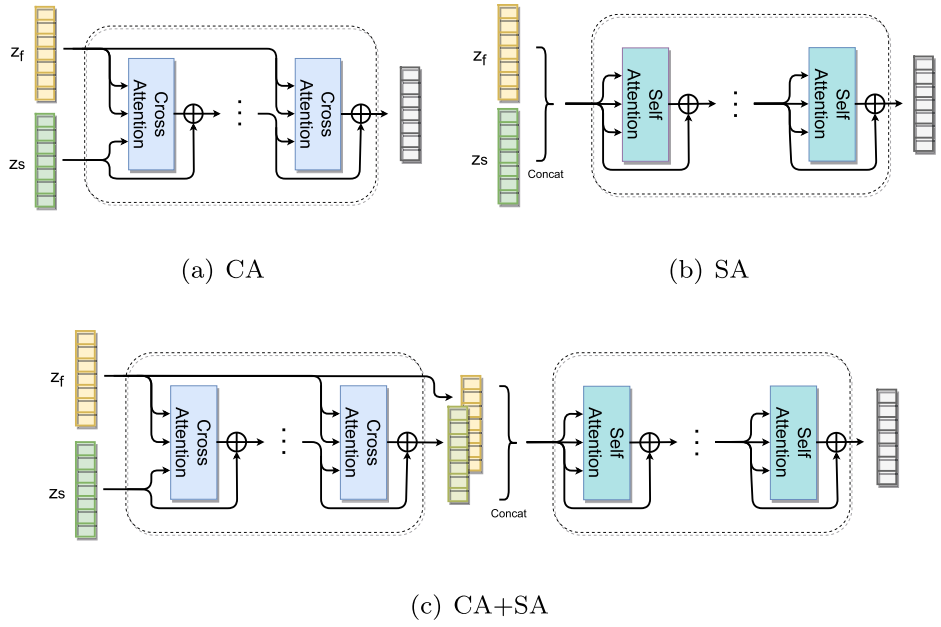

Fig. 11. Demonstration of possible designs for the synthesis module. The architecture in a) only uses cross-attention layers, in b) only uses self-attention layers, and in c) connects these two types of layers, which is used in IHST.

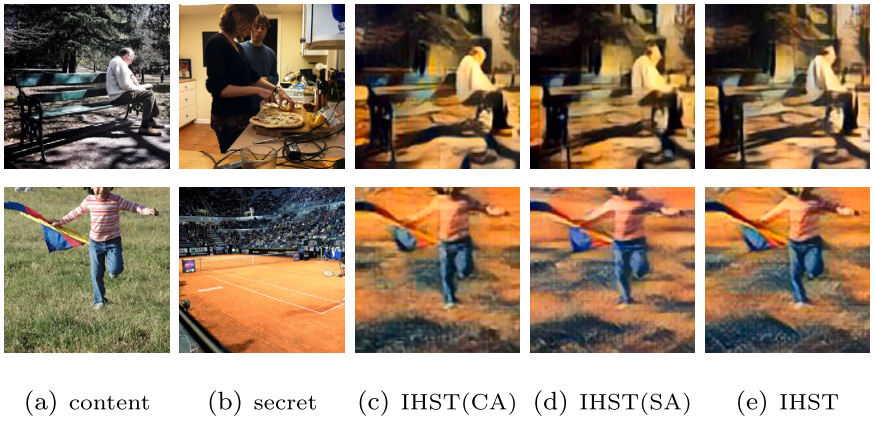

Fig. 12. Target image comparison among IHST with different synthesis modules. Images in a) are the content images, in b) are the secret images, in c), d), and e) are the synthesized images obtained by IHST(CA), IHST(SA), and IHST(CA+SA), respectively.

Figs. 12 and 13 demonstrate the image quality comparison on target and recovered images, respectively. We still use SRNet to evaluate their undetectability, and the detection accuracy during the training process is reported in Fig. 10. We find that self-attention can help synthesize more detail in the target images, while losing some content consistency compared with cross-attention, as shown in the ground in the target images in Fig. 12. On the other hand, the target images generated by cross-attention provide higher recovery accuracy. IHST(CA+SA) outperforms the other two architectures in terms of both image quality and undetectability. As a result, IHST(CA+SA) is selected for our scheme.

4.5.3. Performance on different hyper-parameters

We assess the impact of 4 hyper-parameters in Eq. (15) on the model's performance. To show the advantage of selected hyper-parameters, i.e., $\lambda_1 = 7, \lambda_2 = 1, \lambda_3 = \lambda_4 = 80$, we successively vary each parameter within a range while keeping the others unchanged. The entire network is retrained for each setting. Table 6 compares the recovery accuracies obtained by different hyper-parameter settings. The table shows that the chosen hyper-parameters can achieve a local optimal performance.

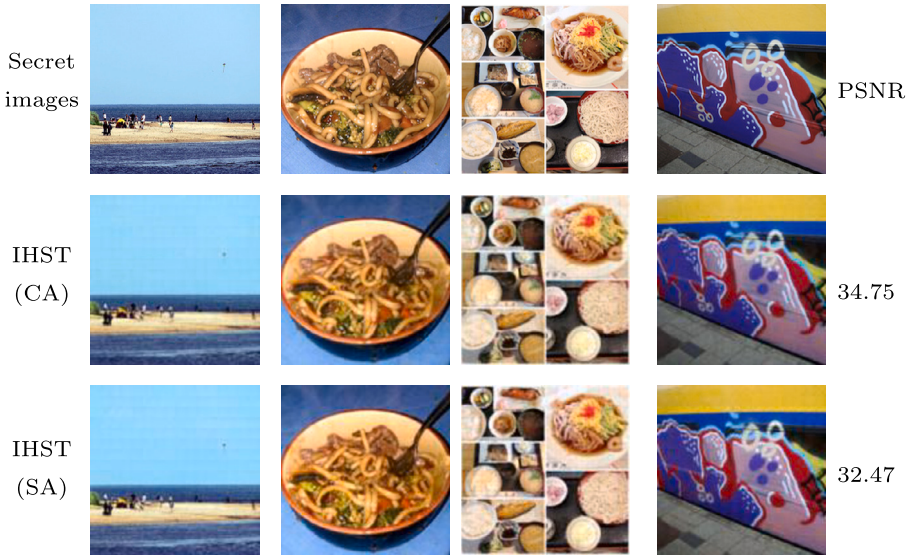

Fig. 13. Recovered image comparison among IHST with different synthesis modules.

Table 6
Recovery accuracy with different loss weights.

λ_1	λ_2	λ_3	λ_4	PSNR	SSIM
7	1	80	80	**35.9**	**0.96**
1	1	80	80	24.9	0.78
4	1	80	80	29.4	0.91
10	1	80	80	32.3	0.94
7	4	80	80	23.4	0.67
7	7	80	80	32.5	0.94
7	10	80	80	30.0	0.92
7	1	10	80	35.3	0.96
7	1	40	80	33.8	0.95
7	1	120	80	24.6	0.73
7	1	80	10	31.1	0.91
7	1	80	40	31.8	0.93
7	1	80	120	30.8	0.93

5. Discussion and conclusion

In this paper, we propose an image hiding network conditional on secret images, named IHST. It stylizes the latent representation in the generator with the style of secret images, which moves the latent representation closer to the secret images and thus improves the recovery accuracy and the security of synthesized target images. There are three modules in IHST: encoding, transfer, and synthesis modules. The transfer module guides the target image transferring to the image domain suitable for embedding. Further, the designed synthesis module can synthesize stego images with high image quality and recovery accuracy. All these modules are trained jointly to improve the style transfer performance as well as recovery accuracy.

We think that there may be two issues when applying the proposed scheme in the real scenario. First, **is it abnormal to transmit synthesized images?** Steganography usually assumes that normal users distribute real-world images. Nevertheless, synthesized images have become popular due to the rapid development of image synthesis and editing tools. Moreover, there are scenarios where the majority of distributed images are synthesized, such as fantasy, comic, etc. As a result, it would be reasonable to choose synthesized images as a public channel. Second, **will the style transfer leak secret information?** It has to be agreed that the style transfer will leak style information of secret images. Consequently, the warder would detect the existence of the secret if he knew the style of secret images to be hidden. Nevertheless, we can transmit stego images in batches to confuse the warder if secret images are distributed similarly to the synthesized images on the public channel.

Experiments have demonstrated that the proposed scheme performs similarly to innocent synthesis tools. It can achieve high quality on target and recovered images simultaneously. Furthermore, despite no adversarial training with steganalyzer or discriminator, the proposed scheme presents high resistance to steganalysis.

CRediT authorship contribution statement

Fenghua Zhang: Methodology, Software, Writing – original draft, Writing – review & editing. **Bingwen Feng:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing. **Zhihua Xia:** Supervision, Validation. **Jian Weng:** Supervision, Writing – review & editing. **Wei Lu:** Supervision, Writing – review & editing. **Bing Chen:** Formal analysis, Validation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No. 61802145, 61932010, 62261160653, 62102101), Natural Science Foundation of Guangdong Province, China (Grant No. 2023A1515011348, 2019B010137005), the Fundamental Research Funds for the Central Universities, the Doctoral Scientific Research Foundation of Guangdong Polytechnic Normal University (No. 2021SDKYA101).

References

[1] N. Provos, P. Honeyman, Hide and seek: an introduction to steganography, IEEE Secur. Priv. 1 (3) (2003) 32–44.
[2] Y. Du, Z. Yin, New framework for code-mapping-based reversible data hiding in jpeg images, Inf. Sci. 609 (2022) 319–338.
[3] T. Filler, J. Judas, J. Fridrich, Minimizing additive distortion in steganography using syndrome-trellis codes, IEEE Trans. Inf. Forensics Secur. 6 (3) (2011) 920–935.
[4] B. Feng, W. Lu, W. Sun, Secure binary image steganography based on minimizing the distortion on the texture, IEEE Trans. Inf. Forensics Secur. 10 (2) (2014) 243–255.
[5] Q. Mao, F. Li, C.-C. Chang, Reversible data hiding with oriented and minimized distortions using cascading trellis coding, Inf. Sci. 317 (2015) 170–180.
[6] S. Baluja, Hiding images within images, IEEE Trans. Pattern Anal. Mach. Intell. 42 (7) (2019) 1685–1697.
[7] Y. Xu, C. Mou, Y. Hu, J. Xie, J. Zhang, Robust invertible image steganography, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 7875–7884.
[8] P. Wei, S. Li, X. Zhang, G. Luo, Z. Qian, Q. Zhou, Generative steganography network, in: Proceedings of the 30th ACM International Conference on Multimedia, 2022, pp. 1621–1629.
[9] C. Zhang, P. Benz, A. Karjauv, G. Sun, I.S. Kweon, Udh: universal deep hiding for steganography, watermarking, and light field messaging, Adv. Neural Inf. Process. Syst. 33 (2020) 10223–10234.
[10] S.-P. Lu, R. Wang, T. Zhong, P.L. Rosin, Large-capacity image steganography based on invertible neural networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 10816–10825.
[11] Z. Guan, J. Jing, X. Deng, M. Xu, L. Jiang, Z. Zhang, Y. Li, Deepmih: deep invertible network for multiple image hiding, IEEE Trans. Pattern Anal. Mach. Intell. 45 (1) (2022) 372–390.
[12] M. Boroumand, M. Chen, J. Fridrich, Deep residual network for steganalysis of digital images, IEEE Trans. Inf. Forensics Secur. 14 (5) (2018) 1181–1193.
[13] G. Xu, H.-Z. Wu, Y.-Q. Shi, Structural design of convolutional neural networks for steganalysis, IEEE Signal Process. Lett. 23 (5) (2016) 708–712.
[14] Z. Zhou, Q.J. Wu, X. Sun, Encoding multiple contextual clues for partial-duplicate image retrieval, Pattern Recognit. Lett. 109 (2018) 18–26.
[15] X. Liu, Z. Li, J. Ma, W. Zhang, J. Zhang, Y. Ding, Robust coverless steganography using limited mapping images, J. King Saud Univ, Comput. Inf. Sci. 34 (7) (2022) 4472–4482.
[16] Q. Li, X. Wang, X. Wang, B. Ma, C. Wang, Y. Shi, An encrypted coverless information hiding method based on generative models, Inf. Sci. 553 (2021) 19–30.
[17] Y. Peng, D. Hu, Y. Wang, K. Chen, G. Pei, W. Zhang, Stegaddpm: generative image steganography based on denoising diffusion probabilistic model, in: Proceedings of the 31st ACM International Conference on Multimedia, 2023, pp. 7143–7151.
[18] G. Li, B. Feng, M. He, J. Weng, W. Lu, High-capacity coverless image steganographic scheme based on image synthesis, Signal Process. Image Commun. (2022) 116894.
[19] Z. Wang, X. Zhang, Z. Qian, Practical cover selection for steganography, IEEE Signal Process. Lett. 27 (2019) 71–75.
[20] K. Zeng, K. Chen, W. Zhang, Y. Wang, N. Yu, Improving robust adaptive steganography via minimizing channel errors, Signal Process. 195 (2022) 108498.
[21] D. Volkhonskiy, I. Nazarov, E. Burnaev, Steganographic generative adversarial networks, in: Twelfth International Conference on Machine Vision (ICMV 2019), vol. 11433, SPIE, 2020, pp. 991–1005.
[22] H. Shi, J. Dong, W. Wang, Y. Qian, X. Zhang, Ssgan: secure steganography based on generative adversarial networks, in: Pacific Rim Conference on Multimedia, 2017.
[23] J. An, S. Huang, Y. Song, D. Dou, W. Liu, J. Luo, Artflow: unbiased image style transfer via reversible neural flows, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 862–871.
[24] X. Liang, B. Liu, Q. Ying, Z. Qian, X. Zhang, Stylestegan: leak-free style transfer based on feature steganography, arXiv preprint arXiv:2307.00225, 2023.
[25] H. Benmeziane, H. Ouarnoughi, K.E. Maghraoui, S. Niar, Real-time style transfer with efficient vision transformers, in: Proceedings of the 5th International Workshop on Edge Systems, Analytics and Networking, 2022, pp. 31–36.
[26] X. Wu, Z. Hu, L. Sheng, D. Xu, Styleformer: real-time arbitrary style transfer via parametric style composition, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 14618–14627.
[27] Y. Deng, F. Tang, W. Dong, C. Ma, X. Pan, L. Wang, C. Xu, Stytr2: image style transfer with transformers, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 11326–11336.
[28] J. Lu, Transformer-based neural texture synthesis and style transfer, in: 2022 4th Asia Pacific Information Technology Conference, 2022, pp. 88–95.
[29] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: Neural Information Processing Systems, 2017.

[30] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805, 2018.
[31] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., An image is worth 16x16 words: transformers for image recognition at scale, arXiv preprint arXiv:2010.11929, 2020.
[32] X. Chen, X. Zheng, K. Sun, W. Liu, Y. Zhang, Self-supervised vision transformer-based few-shot learning for facial expression recognition, Inf. Sci. 634 (2023) 206–226.
[33] P. Esser, R. Rombach, B. Ommer, Taming transformers for high-resolution image synthesis, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 12873–12883.
[34] Y. Jiang, S. Chang, Z. Wang, Transgan: Two transformers can make one strong gan, arXiv preprint arXiv:2102.07074, 2021.
[35] L.A. Gatys, A.S. Ecker, M. Bethge, Image style transfer using convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2414–2423.
[36] J. Johnson, A. Alahi, L. Fei-Fei, Perceptual losses for real-time style transfer and super-resolution, in: European Conference on Computer Vision, Springer, 2016, pp. 694–711.
[37] X. Huang, S. Belongie, Arbitrary style transfer in real-time with adaptive instance normalization, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 1501–1510.
[38] M. Wu, Transforming images into paintings in the style of van gogh based on cyclegan, in: Third International Conference on Intelligent Computing and Human-Computer Interaction (ICHCI 2022), vol. 12509, SPIE, 2023, pp. 313–317.
[39] Z. Wang, N. Gao, X. Wang, J. Xiang, G. Liu, Stnet: a style transformation network for deep image steganography, in: International Conference on Neural Information Processing, Springer, 2019, pp. 3–14.
[40] X. Bi, X. Yang, C. Wang, J. Liu, High-capacity image steganography algorithm based on image style transfer, Secur. Commun. Netw. 2021 (2021) 1–14.
[41] M. Pu, Y. Huang, Y. Liu, Q. Guan, H. Ling, Edter: edge detection with transformer, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 1402–1412.
[42] Z. He, M. Lin, Z. Xu, Z. Yao, H. Chen, A. Alhudhaif, F. Alenezi, Deconv-transformer (dect): a histopathological image classification model for breast cancer based on color deconvolution and transformer architecture, Inf. Sci. 608 (2022) 1093–1112.
[43] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft coco: common objects in context, in: 13th European Conference on Computer Vision (ECCV 2014), 2014, pp. 740–755.
[44] F. Phillips, B. Mackintosh, Wiki art gallery, inc.: a case for critical thinking, Issues Account. Educ. 26 (3) (2011) 593–608.
[45] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, S. Hochreiter, Gans trained by a two time-scale update rule converge to a local nash equilibrium, Adv. Neural Inf. Process. Syst. 30 (2017).
[46] X. Liu, Z. Ma, J. Ma, J. Zhang, G. Schaefer, H. Fang, Image disentanglement autoencoder for steganography without embedding, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 2303–2312.
[47] Y. Wu, W. AbdAlmageed, P. Natarajan, Mantra-Net: Manipulation Tracing Network for Detection and Localization of Image Forgeries with Anomalous Features, 2019, pp. 9543–9552.
[48] B. Boehm, Stegexpose-a tool for detecting LSB steganography, arXiv preprint arXiv:1410.6656, 2014.
[49] X. Weng, Y. Li, L. Chi, Y. Mu, High-capacity convolutional video steganography with temporal residual modeling, in: Proceedings of the 2019 on International Conference on Multimedia Retrieval, 2019, pp. 87–95.