ARTICLE

Robust Information Hiding Based on Neural Style Transfer with Artificial Intelligence

Xiong Zhang[1,2], Minqing Zhang[1,2,3,*], Xu An Wang[1,2,3], Wen Jiang[1,2], Chao Jiang[1,2] and Pan Yang[1,4]

[1]College of Cryptography Engineering, Engineering University of People's Armed Police, Xi'an, 710086, China

[2]Key Laboratory of People's Armed Police for Cryptology and Information Security, Xi'an, 710086, China

[3]Key Laboratory of CTC & Information Engineering (Engineering University of People's Armed Police), Ministry of Education, Xi'an, 710086, China

[4]Staff Department of People's Armed Police Ningxia Corps, Yinchuan, 750000, China

*Corresponding Author: Minqing Zhang. Email: api_zmq@126.com

ABSTRACT

This paper proposes an artificial intelligence-based robust information hiding algorithm to address the issue of confidential information being susceptible to noise attacks during transmission. The algorithm we designed aims to mitigate the impact of various noise attacks on the integrity of secret information during transmission. The method we propose involves encoding secret images into stylized encrypted images and applies adversarial transfer to both the style and content features of the original and embedded data. This process effectively enhances the concealment and imperceptibility of confidential information, thereby improving the security of such information during transmission and reducing security risks. Furthermore, we have designed a specialized attack layer to simulate real-world attacks and common noise scenarios encountered in practical environments. Through adversarial training, the algorithm is strengthened to enhance its resilience against attacks and overall robustness, ensuring better protection against potential threats. Experimental results demonstrate that our proposed algorithm successfully enhances the concealment and unknowability of secret information while maintaining embedding capacity. Additionally, it ensures the quality and fidelity of the stego image. The method we propose not only improves the security and robustness of information hiding technology but also holds practical application value in protecting sensitive data and ensuring the invisibility of confidential information.

KEYWORDS

Information hiding; neural style transfer; robustness

1 Introduction

Information hiding refers to embedding secret information into non secret information to achieve covert communication or embedding copyright information into digital content to achieve copyright protection of digital media. Due to its higher concealment, larger capacity, and wider application scenarios, image information hiding has been widely studied and utilized by academia and industry.

According to the implementation me thods, image information hiding can be divided into traditional image information hiding methods and image information hiding methods based on deep learning [1–4]. Traditional image information hiding usually relies on hand-designed features and models, using classical digital embedding techniques. Examples include Least Significant Bit (LSB) [5], Discrete Cosine Transform (DCT) [6], or Discrete Wavelet Transform (DWT) [7]. These methods are simple to implement and easy to understand. However, their limitation is that some statistical characteristics of the image may be changed [8–10], which increases the possibility of secret communication exposure. The image information hiding method based on deep learning uses Convolutional Neural Networks (CNN) [11] to extract image features and merge them with secret information to generate an image with hidden information. Compared with traditional methods, the image information hiding technology based on deep learning has the characteristics of fast design speed and strong performance improvement ability. However, its embedding capacity and robustness [12,13] still need to be improved.

To solve the above problems, style transfer technology is introduced into information hiding. Style transfer [14,15] refers to extracting the features of the reference style image and the target image using the deep learning model and recombining these features to generate a new image. This allows the new image to have the visual effect of the style image while maintaining the content of the target image. Style transfer is applied to information hiding by embedding the information to be hidden into the style of the image. This takes advantage of the highly nonlinear characteristics of style transfer, mixing the hidden information with the appearance of the image to improve the concealment of the hiding. Additionally, since style transfer preserves the edge, texture, and other features of the image, the hidden information of the image has good visual quality and is difficult to be detected by the attacker.

1.1 Related Works

STNet [16], proposed by Wang et al., can embed secret information in the process of image style transfer. The algorithm combines the steganography process and the style transfer process, showcasing the style transfer function while hiding the steganography function, thereby improving the security of the steganography algorithm.

Building upon this, Bi et al. [17] proposed ISTNet. By improving the decoder of STNet, the secret image features were fused with the results of the Multi-scale Adaptive Instance Normalization (AdaIN) [18] layer, and the gray image was hidden in the process of image style transmission to improve the steganography ability.

AdaIN is a scalable and effective image style transfer technique that has been widely used in image style transfer, image enhancement, and image generation. This approach allows for the application of one image's style to another, creating images with different styles.

The structure of AdaIN is depicted in Fig. 1, where the input image undergoes feature extraction through a convolutional neural network. Subsequently, the features are normalized by calculating the mean and variance of each channel. Similarly, the features of the reference image are normalized during style transfer. The mean and variance of the input image's features are then adjusted to align with those of the reference image. Finally, the adjusted features pass through a deconvolutional neural network to generate the transformed image.

To overcome the dependence of the algorithm on pre-trained CNNs, Hu et al. [19] proposed an end-to-end steganography algorithm without pre-training, which embedded the secret message while converting the style of the content image.
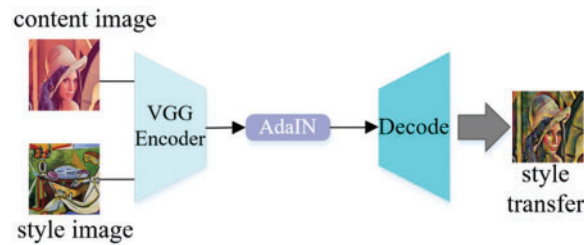
Figure 1: AdaIN model diagram

Aiming at the problem of color image embedding, Yang et al. [20] extracted the semantic information of the carrier image, the style information of the style image, and the feature information of the color image based on the style transfer process and the feature extraction advantages of the convolutional neural network. They fused these together and realized the style transfer of the carrier image and the embedding of the color image through the decoder.

1.2 Objective of This Study

Specifcally, thenovelties and contributions of this work are three-fold:

- In order to better ensure the security of information during transmission, this paper designs an information hiding algorithm based on neural style transfer. By stylizing and disguising stego-images, attackers cannot visually determine whether they are stego-images, thereby improving the security of stego-images.
- This paper designs an attack layer in the network structure to simulate attacks and noise from real environments. Through adversarial training, the algorithm's anti attack ability is improved, making the transmission of stego-images more secure, thereby improving the security and accuracy of secret information.
- This paper introduces the Inception module into the network structure, which uses multiple convolutional kernels of different sizes at the same level to extract image features. This improves the model's expressive power, reduces computational costs, and enhances the algorithm's anti attack ability and extraction accuracy.

The algorithm in this paper aims to ensure the secure transmission of secret information on public channels, and it can improve the concealment and unknowability of secret information while ensuring the quality and fidelity of stego-images. It has certain application value for improving the robustness of information hiding.

2 The Proposed Scheme

The current image information hiding methods based on style transfer face a challenge in balancing embedding capacity and robustness simultaneously. To address this, a robust information hiding algorithm based on style transfer is proposed in this paper. This algorithm utilizes a deep learning model to extract features from the reference style image and the content image, reorganizing these features to generate a new image while preserving the content of the target image. The addition of an Inception Block on the basis of convolution enhances the model's expressive capability and accuracy. Furthermore, to ensure high embedding capacity while considering robustness, an attack layer is designed to simulate various real-world attacks. Through joint training of the hidden network

and the extraction network, the extractor's ability to extract the secret image is enhanced, thereby improving the algorithm's robustness.

2.1 Model Framework

The overall structure of the proposed algorithm, as shown in Fig. 2, comprises three main parts: The hidden network, noise attack layer, and extraction network. The hiding network includes a feature extraction network and a decoder network. It combines the style image's characteristics to transform the content image into an artistic image with the style image's features, embedding the information of the secret image to generate the stego image. The attack layer simulates various real-world attacks, while the extraction network facilitates the extraction of secret images from stego images.
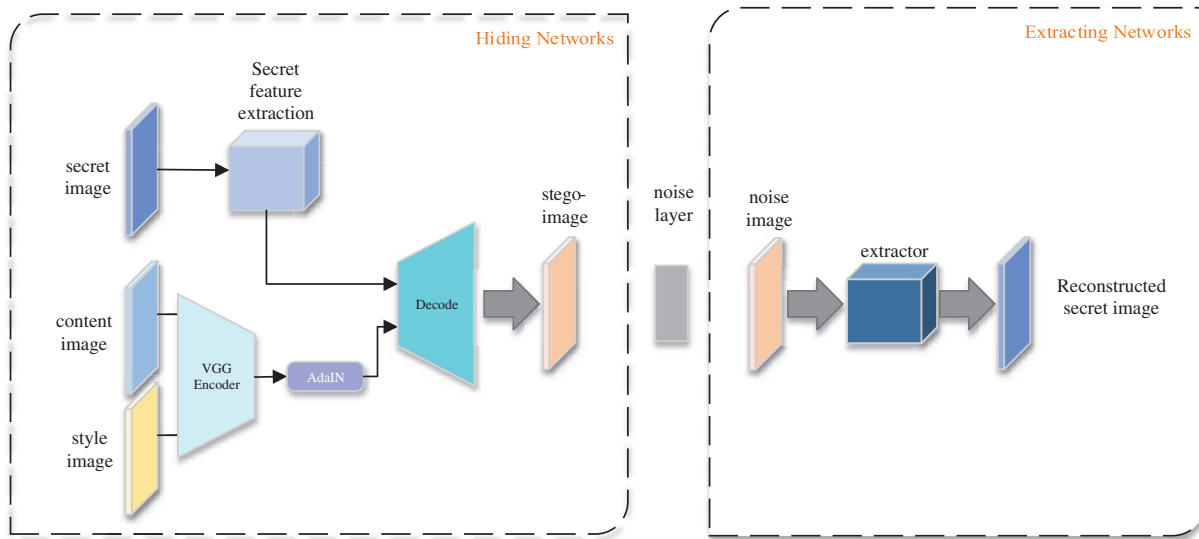

Figure 2: Overall framework of robust information hiding algorithm based on neural style transfer

2.1.1 Hiding Networks

The hiding network comprises two components: A feature extraction network and a decoder network. Leveraging the convolutional neural network's efficacy in image tasks, the feature extraction network predominantly employs the VGG [21] convolutional neural network. This network sequentially extracts features from input images layer by layer. Enhancing training efficacy, we utilize the pre-trained VGG model in PyTorch, trained on the ImageNet dataset. Employing a pre-trained feature extraction network reduces training overhead and time, yielding superior feature extraction outcomes. Inputting any content image and style image into the hidden network results in the generation of a stylized image that amalgamates the content of the former with the style of the latter. That is, the content feature (f_c) of the content image and the style feature (f_s) of the style image are obtained. In the proposed algorithm, f_c and f_s are encoded into a feature through the AdaIN layer, as shown in Eq. (1), which is combined with the same size feature of the secret image to achieve style fusion. It is then input to the decoder, through which the obtained feature is mapped back to the image space.

$$T = \text{AdaIN} = \mu\left(f_s\right) + \sigma\left(f_s\right)\frac{f_c - \mu\left(f_c\right)}{\sigma\left(f_c\right)} \tag{1}$$

where T is the output content of AdaIN, f_c is the content feature of the content image, f_s represents the feature information of the style image, μ represents the mean value, and σ represents the standard deviation.

The feature extraction network employed in this study consists primarily of convolutional networks and Inception [22] structures. The Inception module, depicted in Fig. 3, utilizes multi-channel convolution, employing multiple convolution kernels of different sizes at the same level for feature extraction. This approach improves the model's expressive power to a certain extent. Additionally, it mitigates the issues caused by excessive parameters and lengthy computation time when using a large number of convolution kernels simultaneously for dimension reduction and expansion. The computational cost of the model is significantly reduced, while the feature information of the secret image is maximized. This, in turn, facilitates subsequent resistance against various attacks from the attack layer, ultimately enhancing the accuracy of secret extraction within the extraction network.
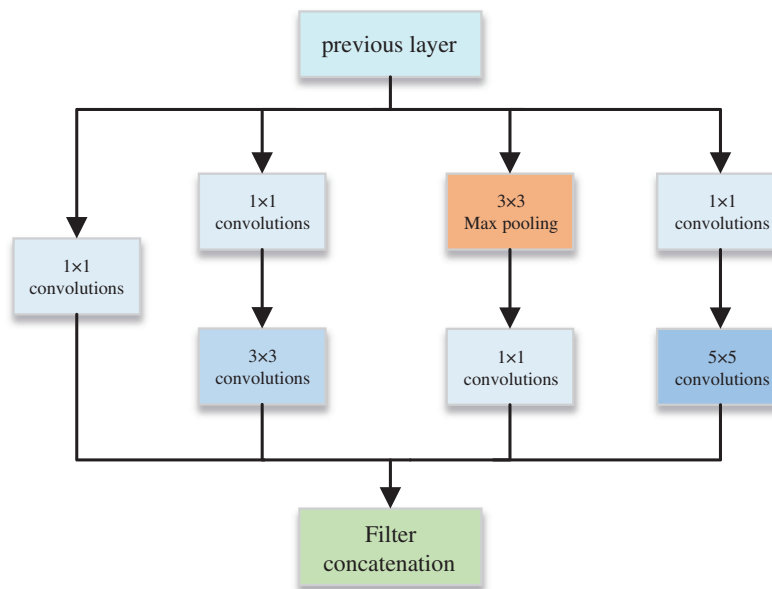

Figure 3: Inception block structure

The input feature information is decoded by the decoder to generate the stylized stego-image, as depicted in Fig. 4. Reflectionpad is used for mirror filling, and ReLU is utilized as the activation function.

2.1.2 Attack Layer

Image attacks are prevalent in real-world scenarios. To enhance the practical applications of information hiding technology, an attack layer is incorporated into the network in this paper. We propose a parameterless noise layer that introduces distortion in the encoded image, resulting in a noisy image. The network is trained by introducing noise, and the attack image is employed for adversarial training, following a similar approach to HiDDeN [1], Common digital attacks are simulated using differentiable methods. The decoder then generates the predicted message from the noisy image. We demonstrate the simulation of dimension-reducing noise, such as cropping, without requiring the encoded image and the noisy image to have identical dimensions. Various attacks, including JPEG

compression, crop, and Gaussian noise, are applied to the stego-image. Each non-identity noise layer is associated with a scalar hyperparameter to regulate the intensity of distortion: Crop retains a small portion of the pixels, Gaussian has a kernel width σ, and JPEG has a quality factor $Q \in (0,100)$.
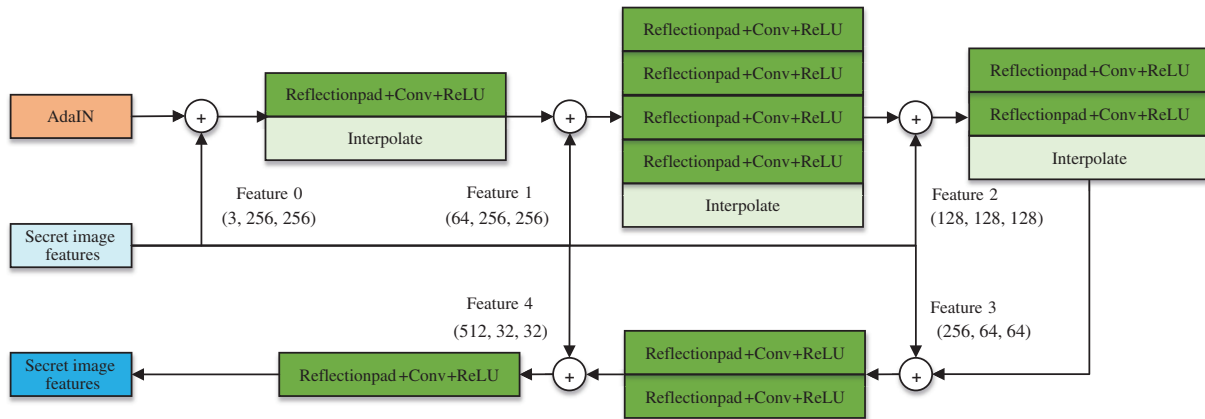

Figure 4: Decoder structure

To limit the information transmitted through specific frequency domain channels, we have developed noise layers that mimic JPEG compression. These layers utilize 8×8 stride 8 convolutional layers to implement the DCT transformation, with each filter corresponding to a basis vector in the DCT transform. As a result, the network activations represent the DCT domain coefficients of the encoded image. Subsequently, masking is applied to the DCT coefficients to constrain information flow. Transposed convolution is then used to perform inverse DCT transformation and generate the noisy image. JPEG-Mask applies a fixed mask, retaining only 25 low-frequency DCT coefficients in the one channel and 9 coefficients in the other two channels. Other coefficients are set to zero. This approach effectively produces models robust to actual JPEG compression.

During training, four models are trained using different noise layers: Crop ($p = 0.035$), Gaussian ($\sigma = 2.0$), JPEG-mask and JPEG-drop. These noise-resistant models are known as specialized models, as they are trained to withstand specific types of noise. Additionally, a noise-resistant combination model is trained by employing different noise layers for each mini-batch during training.

All models are trained on 10,000 cover images from the COCO dataset, adjusted to fit the dimensions of the specific experiment. Evaluation is conducted on a test set of 1000 images that were not used during training. Message sampling involves random bit extraction. For gradient descent, we utilize Adam with a learning rate of 0.001 and default hyperparameters. All models are trained in batches of 10, with the training lasting for 100 epochs.

2.1.3 Extracting Networks

The primary objective of the extraction network is to retrieve the secret image from the stego image, enabling the recovery of the concealed content. During model training, both the extraction network and the hidden network can be trained jointly. The stego-image, generated by the hidden network, undergoes various types of noise attacks to produce the attacked stego image. Upon receiving the noise-attacked stego-image, the extraction network conducts a series of convolution operations to extract the secret image and uncover the hidden message. The network structure is depicted in Fig. 5. During convolution operations, the default convolution kernel size is 3×3 with a stride of 1.
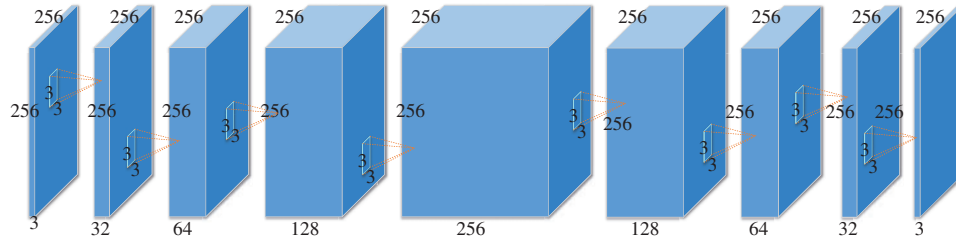
Figure 5: Extracting network neural structure diagram

2.2 Loss Function

The overall model's loss function comprises three components: The content loss function (Lc), the style loss function (Ls), and the loss function (Le) for the secret image. The content loss is defined as the Euclidean distance between the target feature and the output image feature. The style loss is calculated based on the mean and standard deviation of the feature maps from both the style image and the hidden image. To achieve faster convergence and preserve the content features of the carrier image, the output T from AdaIN is utilized as the target feature to constrain the features of the generated image.

The content loss function Lc aims to constrain the generated stego image and preserve its content information. Thus, the loss function is represented by Eq. (2).

$$Lc = \|f_m - t\|_2 \tag{2}$$

The style loss function Ls primarily measures the degree of stylistic similarity between the style of the secret-containing image and the style image. Initially, the encoder is employed to acquire the feature maps of the secret image features f_m and the style image (s) in different layers. Subsequently, the mean and variance of these feature maps are computed. Ls is calculated as the sum of Euclidean distances between the mean and the variance, as depicted in Eq. (3).

$$Ls = \frac{1}{n} \sum_{i=1}^{n} \left(\begin{array}{c} \mu\left(\phi_i\left(f_m\right)\right) - \mu\left(\phi_i\left(s\right)\right)_2^2 \\ + \sigma\left(\phi_i\left(f_m\right)\right) - \sigma\left(\phi_i\left(s\right)\right)_2^2 \end{array} \right) \tag{3}$$

As illustrated in Eq. (4), $\|\cdot\|$ represents the norm, A denotes the discriminator, G denotes the generator, N denotes the noise vector, and L_G represents the generator's loss function. The generator's objective is to induce the discriminator to classify the generated samples as real ones. The loss function L_G is computed by summing the squared errors of the discriminator outputs.

$$LG = \frac{1}{2}\|A\left(G\left(N\right)\right) - 1\|^2 \tag{4}$$

To enhance the recovery effect of the secret image, the reconstruction loss function, Le, primarily measures the disparity between the original image and the extracted image. Here, g represents the distance between secret image (s) and extracted image (s'). The secret loss is expressed in Eq. (5), α as a hyper-parameter.

$$Le = g\left(s, s'\right) + \alpha LG \tag{5}$$

The overall loss function is a weighted combination of content loss, style loss, and secret image loss. The weight factors, β and γ, control the significance of each loss term, as depicted in Eq. (6).

$$L = Lc + \beta Ls + \gamma Le \tag{6}$$

3 Experimental Results and Analysis

The experimental platform utilizes PyTorch 1.9, employs Python 3.7 as the programming language, and utilizes an RTX3080 graphics card with 12 GB of video memory. The dataset for the experimental secret image comprises the COCO number set, and the sample dataset is the ukiyoe2photo dataset. The size of each image is uniformly adjusted to 256×256, and the hyper-parameters are set to $\alpha = 0.05$, $\beta = 10$, $\gamma = 10$.

The algorithm proposed in this paper represents a robust improvement of the information hiding algorithm based on neural style migration. The primary focus is to enhance the utility of the algorithm in lossy channels.

In the comparison of image stylistic migration effect, the addition of the attack layer does not impact the stylistic effect of the stego-image. The robustness enhancement is primarily achieved by adding the attack layer after generating the stego-image, by which time the stylistic effect of the image has already been generated.

3.1 The Visual Effect of the Image

The visual effect of an image serves as the most fundamental and intuitive evaluation index for information hiding algorithms. Four groups of images, including content images, style images, secret images, and stego-images along with extracted images, are randomly selected from the experimental results, as illustrated in Fig. 6. And it demonstrates the algorithm's improved capability in achieving style migration and embedding secret information.
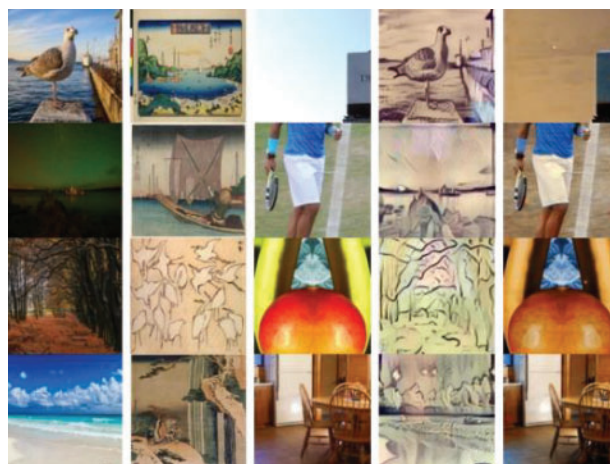
(a) Content (b) Style (c) Secret (d) stego- (e) Extracted
  image   image   image   image   image

Figure 6: Image effect of the algorithm

3.2 The Quality of the Stego Image

To assess the quality of dense-containing images, a set of images subjected to various noise attacks is randomly chosen for residual analysis, depicted in Fig. 7.
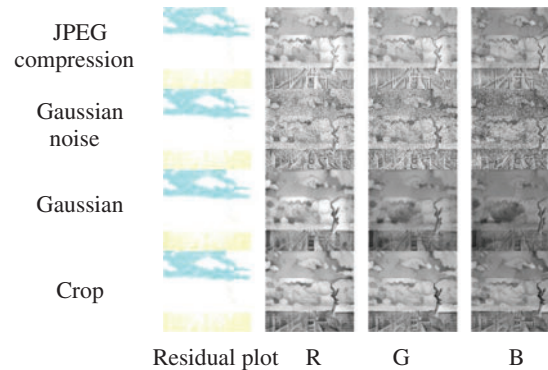

Figure 7: Residual analysis of encrypted images under different attacks

The residual information is then separated into the three color channels: Red (R), green (G), and blue (B). From Fig. 7, it is evident that the differences between the images in the three channels are minimal. This suggests that the image color is well-balanced, devoid of noticeable color deviation or brightness disparity. The details are more intact, without obvious loss or blurring. The image exhibits higher color accuracy and superior color reproduction. Therefore, it can be inferred that the density-containing image exhibits better quality.

3.3 Robustness Testing

To validate the quality of the extracted images, 200 sets of embedding experiments are conducted. Secret images are extracted after subjecting the 200 generated stego-images to various attacks, as depicted in Fig. 8. The extracted secret images are compared with the original ones.
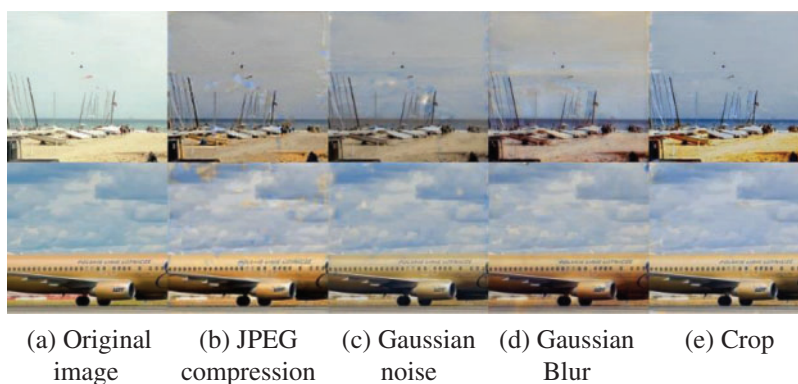

| (a) Original image | (b) JPEG compression | (c) Gaussian noise | (d) Gaussian Blur | (e) Crop |

Figure 8: Comparison of extracting images under different attacks

In this paper, Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity (SSIM) serve as measures to evaluate the efficacy of extracted secret images. A consistent noise attack is applied 1–100 times to the same image. The PSNR and SSIM values of the extracted image post-attack are illustrated in Fig. 9. The overall stability of the extracted images remains consistent, albeit with a

decline in quality compared to the no-attack condition. The PSNR values are roughly distributed around 30 dB, indicating that the distortion loss of image quality is within an acceptable range. From the distribution of SSIM values, it can be seen that our algorithm can effectively resist attacks from Gaussian noise.
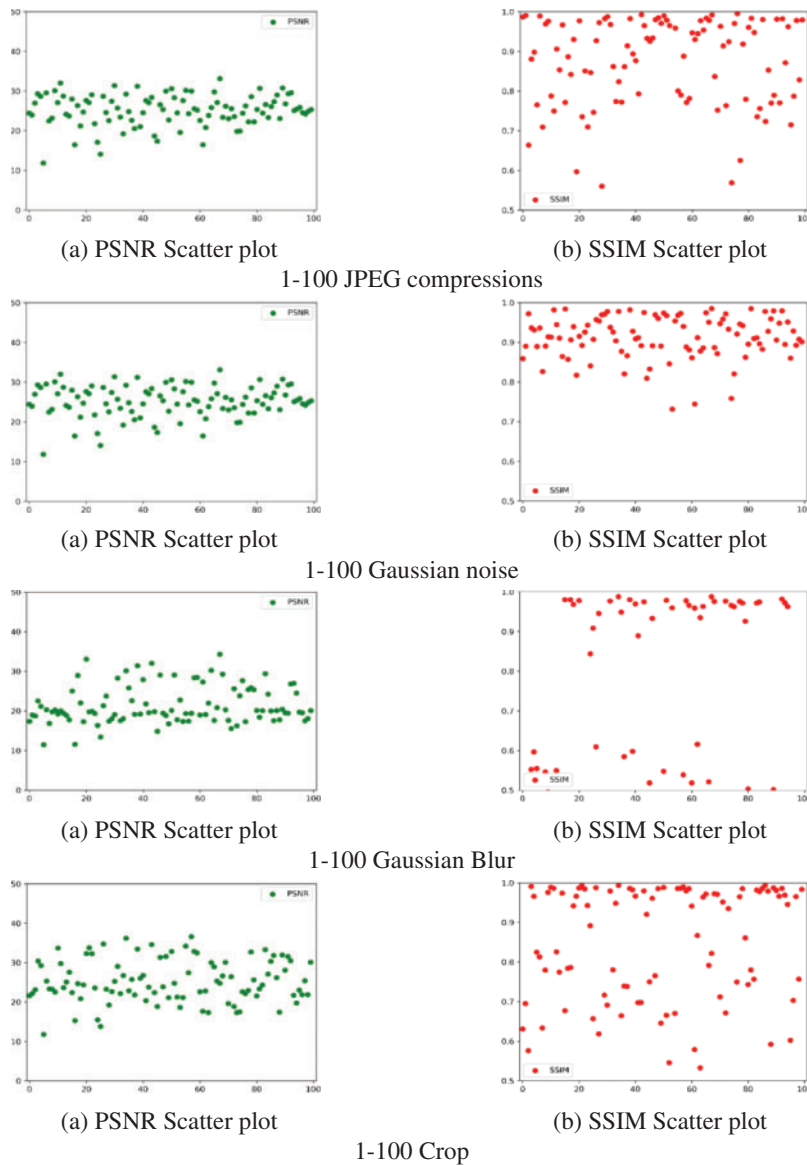
(a) PSNR Scatter plot	(b) SSIM Scatter plot

1-100 JPEG compressions

(a) PSNR Scatter plot	(b) SSIM Scatter plot

1-100 Gaussian noise

(a) PSNR Scatter plot	(b) SSIM Scatter plot

1-100 Gaussian Blur

(a) PSNR Scatter plot	(b) SSIM Scatter plot

1-100 Crop

Figure 9: Scatter plots for extracting image quality after different attacks

Additionally, the algorithm presented in this paper is compared with other robust information hiding algorithms, as illustrated in Tables 1 and 2.

Table 1: Comparison of PSNR of different algorithms

Algorithm	Original image	JPEG compression	Crop	Gaussian blur
[17]	30.48	/	/	/
[23]	25.22	17.64	17.41	14.63
[24]	29.76	26.69	28.47	28.32
Ours	27.81	25.42	24.635	25.31

Table 2: Comparison of SSIM for different algorithms

Algorithm	Original image	JPEG compression	Crop	Gaussian
[17]	0.75	/	/	/
[23]	0.867	0.634	0.625	0.498
[24]	0.916	0.875	0.895	0.905
Ours	0.970	0.874	0.823	0.913

Tables 1 and 2 reveal that, comparing to similar stylised relocation-based information hiding algorithms [17], the SSIM in this paper is significantly larger than that in the literature [17], albeit with a slightly smaller PSNR value. When compared with ADH-GAN [23] and Ying et al. [24], the image quality of the extracted images outperforms that of literature [23], but certain metrics are inferior to those of Ying et al.'s algorithm. The disparity arises primarily because Ying et al.'s algorithm gradually recovers the secret image through the residuals of the contained image and the carrier image. In contrast, the algorithm presented in this paper is tested by attacking the stylised contained image, which precludes the use of gradual recovery for image extraction. Instead, it adopts direct extraction. Additionally, both style conversion and noise attack result in damage to the secret image, making extraction more challenging.

3.4 Analysis of Embedding Capacity

Regarding hiding capacity, the algorithm presented in this paper embeds a 256×256 color image into a 256×256 style image. In other words, each pixel can conceal 24 bits, resulting in a hiding capacity of 24 bpp.

As depicted in Table 3, the algorithm presented in this paper exhibits a significantly larger embedding capacity when compared with similar style migration-based information hiding algorithms in literature [16] and literature [17]. In literature [23], to ensure the robustness and imperceptibility of the target image, the maximum embedded secret pixel ratio should be designed to be less than 0.5. In literature [24], while the embedding capacity for a single secret image is 24 bpp, the proposed method involves embedding multiple secret images. Based on this, in subsequent stages, the embedding capacity is expected to surpass 24 bpp when embedding multiple secret images. In literature [24], while the embedding capacity for a single secret image is 24 bpp, the proposed method involves embedding multiple secret images. The embedding capacity for multiple secret images is expected to surpass 24 bpp.

Table 3: Comparison of embedding capacity of different algorithms

Algorithm	Content image (pixel)	Secret image (bit)	Capacity (bpp)
[16]	Arbitrary	Arbitrary	0.4
[17]	256×256	$256 \times 256 \times 8$	8
[23]	256×256	$<256 \times 128 \times 24$	<12
[24]	256×256	$256 \times 256 \times 24$	24
Ours	256×256	$256 \times 256 \times 24$	24

3.5 Security Analysis

The secure transmission of confidential images holds paramount importance in the realm of information hiding. Unlike traditional information hiding algorithms, which focus on detecting whether an image contains concealed information, steganalysis in neural style migration-based techniques faces the unique challenge of distinguishing between stylized images and confidential ones. This is because the stylization process inherently serves as a form of protection for the confidential image. By altering the pixels of the image, the stylization effectively masks any modifications made to the confidential content, rendering steganalysis significantly more challenging.

In addition, during the robustness testing, it was demonstrated through residual analysis that the noise attack does not pose a threat to the security of the secret image. Therefore, the generated image containing the secret image can be deemed secure.

4 Conclusion

This paper proposes a robust information embedding scheme for style migration based on neural networks. The scheme achieves color secret image stylization and enhances the algorithm's attack layer. This enhancement effectively withstands noise attacks in real environments with lossy channels, thereby improving the robustness of the hiding algorithm through training. Experimental results demonstrate that, compared to other robust information hiding algorithms, this paper enhances algorithm robustness while ensuring large capacity and security.

Additionally, experiments on concealing multiple images within one image, while ensuring robustness, it should be conducted in the next phase.

Author Contributions: The authors confirm contribution to the paper as follows: Study conception and design: X. Zhang, M. Zhang, X. Wang; data collection: W. Jiang, C. Jiang; analysis and interpretation

of results: P. Yang; draft manuscript preparation: X. Zhang. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Not applicable.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] J. Zhu, R. Kaplan, J. Johnson, and F. L. Hidden, "Hiding data with deep networks," in *Proc. ECCV*, Munich, Germany, 2018, pp. 682–697.

[2] J. C. Hou, B. Ou, H. W. Tian, and Z. Qin, "Reversible data hiding based on multiple histograms modification and deep neural networks," *Signal Process.: Image Commun.*, vol. 92, no. 2, pp. 116118, 2021. doi: 10.1016/j.image.2020.116118.

[3] L. Liu, L. Meng, Y. Peng, and X. Wang, "A data hiding scheme based on U-Net and wavelet transform," *Knowl. Based Syst.*, vol. 223, no. 6, pp. 107022, Apr. 2021. doi: 10.1016/j.knosys.2021.107022.

[4] N. R. Kumar, R. B. Krishnan, G. Manikandan, V. Subramaniyaswamy, and K. Kotecha, "Reversible data hiding scheme using deep learning and visual cryptography for medical image communication," *J. Electron. Imaging*, vol. 31, no. 6, pp. 63028, Nov. 2022. doi: 10.1117/1.JEI.31.6.063028.

[5] J. Fridrich, M. Goljan, and R. Du, "Detecting lsb steganography in color, and gray-scale images," *IEEE Multimed.*, vol. 8, no. 4, pp. 22–28, Nov. 2001. doi: 10.1109/93.959097.

[6] J. Huang, Q. Yun, and W. Cheng, "Image watermarking in DCT: An embedding strategy and algorithm," (in Chinese), *Acta Electonica Sinica*, vol. 28, no. 4, pp. 57, Apr. 2000.

[7] L. Zhang and Q. Zheng, "Audio steganography algorithm based on lifting wavelet transform and matrix coding," (in Chinese), *Comput. Appl.*, vol. 29, no. 11, pp. 2942–2945, Dec. 2009. doi: 10.3724/SP.J.1087. 2009.02942.

[8] T. Bhuiyan, A. H. Sarower, R. Karim, and M. Hassan, "An image steganography algorithm using LSB replacement through XOR substitution," in *Proc. IEEE ICOIACT*, Aug. 2019, pp. 44–49.

[9] X. Liao, J. Yin, M. Chen, and Z. Qin, "Adaptive payload distribution in multiple images steganography based on image texture features," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 2, pp. 897–911, 2020. doi: 10.1109/TDSC.2020.3004708.

[10] X. Liao, Y. Yu, B. Li, Z. Li, and Z. Qin, "A new payload partition strategy in color image steganography," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 3, pp. 685–696, 2019. doi: 10.1109/TCSVT.2019.2896270.

[11] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. doi: 10.1109/5.726791.

[12] Z. Zhou, Y. Cao, M. Wang, E. Fan, and Q. M. J. Wu, "Faster-rcnn based robust coverless information hiding system in cloud environment," *IEEE Access*, vol. 7, pp. 179891–179897, Nov. 2019. doi: 10.1109/ACCESS.2019.2955990.

[13] G. Si, C. Qin, and H. Yao, "Robust coverless data hiding based on texture classification and synthesis," (in Chinese), *J. Appl. Sci.*, vol. 38, no. 3, pp. 441–454, 2020. doi: 10.3969/j.issn.0255-8297.2020.03.010.

[14] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. CVPR*, Las Vegas, Nevada, 2016, pp. 2414–2423.

[15] Y. Jing, Y. Yang, Z. Feng, J. Ye, Y. Yu and M. Song, "Neural style transfer: A review," *IEEE Trans. Vis. Comput. Graph*, vol. 26, no. 11, pp. 3365–3385, 2019. doi: 10.1109/TVCG.2019.2921336.

[16] Z. Wang, N. Gao, X. Wang, J. Xiang, and G. Liu, "STNet: A style transformation network for deep image steganography," in *Proc. ICONIP*, Sydney, NSW, Australia, 2019, pp. 3–14.

[17] X. Bi, X. Yang, C. Wang, and J. Liu, "High-capacity image steganography algorithm based on image style transfer," *Secur. Commun. Netw.*, vol. 2021, pp. 1–14, 2021. doi: 10.1155/2021/4179340.

[18] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proc. ICCV*, Venice, Italy, 2017, pp. 1501–1510.

[19] D. Hu, Y. Zhang, C. Yu, J. Wang, and Y. Wang, "Image steganography based on style transfer," arXiv preprint arXiv:2203.04500, 2022.

[20] P. Yang, M. Zhang, Y. Ge, and Y. Zhang, "Color image information hiding algorithm based on style transfer process," *Comput. Appl.*, vol. 46, no. 6, pp. 1730–1735, 2023.

[21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.

[22] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. CVPR*, Boston, USA, 2015, pp. 1–9.

[23] C. Yu, "Attention based data hiding with generative adversarial networks," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 1, pp. 1120–1128, 2020. doi: 10.1609/aaai.v34i01.5463.

[24] Q. Ying, H. Zhou, X. Zeng, H. Xu, Z. Qian and X. Zhang, "Hiding images into images with real-world robustness," in *Proc. ICIP*, Tel Aviv, Israel, 2022, pp. 111–115.