The Visual Computer (2024) 40:5857–5870
https://doi.org/10.1007/s00371-023-03140-1

ORIGINAL ARTICLE

Visual image encryption based on compressed sensing and Cycle-GAN

Zhaoyang Liu[1,2,3] · **Ru Xue**[1,2,3]

Accepted: 13 October 2023 / Published online: 15 November 2023

Abstract

At present, most image encryption schemes directly change plaintext images into ciphertext images without visual significance, and such ciphertext images can be detected by hackers during transmission, and therefore subject to various attacks. To protect the content security and visual safety of images, a learning visual image encryption scheme based on compressed sensing (CS) and cycle generative adversarial network is proposed. First, the secret image is sparse by discrete wavelet transform and compressed by CS. Secondly, the compressed image is permuted and diffused by an improved Henon map to obtain the ciphertext image. Finally, the images are migrated from the ciphertext domain to the plaintext domain by generating an adversarial network to obtain visually meaningful images. We constrain and guide the image generation process by introducing a feature loss function to guarantee the quality of the reconstructed images. Experimental results and security analysis show that the image encryption scheme has sufficient key space, strong key sensitivity, and high reconstruction quality.

Keywords Discrete wavelet transform · Compressed sensing · Cycle generative adversarial network · Improved Henon map · Visual image encryption

1 Introduction

With the rapid development of computer technology, the ability to process, transmit, and store information has increased, but it also poses security risks. Information may be exposed to the risk of theft and leakage during transmission and storage, which may lead to the use of information for illegal purposes, such as attacks against political, military, commercial, and personal identity. If information is tampered with, it can lead to misunderstandings, wrong decisions, losses, and other problems. Information security is important for organizations, businesses, and individuals to protect privacy, ensure confidentiality, maintain credibility, and enhance security and compliance. With the rapid development of network technology, information security has become a global issue that everyone should pay attention to and be concerned about. Currently, one of the most effective methods to ensure the security of information is image encryption technology, such as optical encryption [1–3], chaotic encryption [4, 5], and neural network encryption [6, 7]. However, various image encryption methods are proposed by different authors that convert the plain image to an unintelligible image that looks noisy to maintain confidentiality. From an attacker's perceptive, these unintelligible noisy images indicate that something important is transmitted [8].

In order to achieve visual security while protecting data security, encryption technology, and information hiding technology can be combined to achieve complementary advantages. Cryptography is a method of converting data into a completely unreadable form, which prevents adversaries from identifying the original digital media. However, data hiding techniques can embed secrets into the cover media in an imperceptible manner so that adversaries are unaware of its existence. Bao et al. [9] first introduced this concept of visually meaningful image encryption (VMIE), meaning encrypting a plaintext image and embedding it into a cover image to achieve both content and visual protection. Subsequently, VMIE has attracted the attention of a growing number of researchers. In reference [10], a multi-image visual encryption algorithm based on compression-aware and Schur decomposition is proposed. The compressed image

✉ Ru Xue
rxue@xzmu.edu.cn

1 School of Information Engineering, Xizang Minzu University, Xianyang 712082, Shaanxi, China

2 Key Laboratory of Optical Information Processing and Visualization Technology of Tibet Autonomous Region, Xianyang 712082, Shaanxi, China

3 Xizang Cyberspace Governance Research Center, Xianyang, China

is encrypted and decomposed to embed into the medium and high-frequency sub-bands of the carrier image, which improves the encryption efficiency and security. In reference [11], an encryption scheme that combines half-tensor product compression perception and block substitution techniques is proposed. Block pair substitution of two image blocks with similar standard deviation values makes the cipher image more similar to the carrier image, improving the embedding capability and the visual quality of the image. Ren et al. [12] proposed a visually secure image encryption scheme based on fractional-order chaotic systems and CS techniques. The visual security of cryptographic images is improved by using smoothing functions and coefficient quantization. The use of compression awareness effectively improves transmission efficiency and provides a degree of protection against plaintext attacks [13, 14]. However, most of them have limited embedding capacity and the embedding process changes the image structure which cannot effectively resist steganalysis techniques. More importantly, their embedding and corresponding extraction operations are not fully reversible, i.e., there is a loss of energy, causing poor quality of the reconstructed images.

Recently, image encryption combined with deep learning has been widely studied in reference [15], which uses deep neural networks to generate cryptographic images directly without training the network, and the weights of the network are controlled by the DCT coefficients of the garbled code. The nonlinearity in the encryption scheme due to the multilayer and activation functions makes it resistant to attacks. Reference [16] used an iris image as the key for an image encryption scheme. The deep learning model extracts the feature vectors of the iris image and encodes them using RS error correction codes to perform encryption, which improves the encryption efficiency. Reference [17] combines Cycle-GAN with the traditional diffusion algorithm to enhance the avalanche effect and also effectively break the adjacent pixel correlation of the image. Thus, it can be seen that deep learning has revolutionized cryptographic systems, not only improving the efficiency of encryption but also greatly enhancing security [18].

Therefore, a visual image encryption scheme based on CS and Cycle-GAN is proposed. First, the plaintext image is sparse and compressed with CS using discrete wavelet transform (DWT); then, the ciphertext image is obtained by permutation and diffusion through the improved Henon map. Finally, The Cycle-GAN deep learning network accomplishes the migration from the ciphertext domain to the plaintext domain and produces a visual image. Compared with the traditional VMIE, the secret image is not embedded in any carrier image; therefore, it can effectively resist steganalysis.

Overall, the advantages are summarized as follows:

(1) The use of generative adversarial networks for visual image encryption enables image translation from the ciphertext domain to the plaintext domain, protecting data security and ensuring visual security at the same time;
(2) CS and chaotic map can improve transmission efficiency while ensuring security, and the use of deep learning networks greatly increases the key space;
(3) We introduce a feature loss function to constrain and guide the image generation, which preserves the detailed features and greatly improves the reconstructed image quality.
(4) In our paper, the domain migration property of cycle generative adversarial networks is used for the first time to visualize encryption. Compared with visual encryption schemes that embed images into cover images, our scheme improves transmission efficiency and resists steganalysis effectively.

The remaining part is as follows. Relevant knowledge is introduced in Sect. 2. Section 3 describes the whole algorithm. Simulation results and security analysis are given in Sect. 4. Section 5 concludes.

2 Relevant knowledge

2.1 Compressive sensing

Compressed sensing [19] is a new type of signal sampling and reconstruction technique developed in recent years. Its core idea is to take advantage of signal sparsity or low-rank nature to obtain low-dimensional information with high probability through nonlinear measurement and sparse representation techniques. It can greatly reduce the amount of signal or image sampling, and improve the efficiency and reliability of data acquisition and transmission while maintaining the image quality or data reconstruction error. Specifically, for a real signal $x \in R^N$, if it is K-sparse under some $N \times N$ orthogonal basis Ψ, i.e., when there are K ($K \ll N$) nonzero elements in s, a one-dimensional measurement y of length M can be obtained by using a measurement matrix Φ of size $M \times N$ ($M < N$) for the observation of signal$x$:

$$y = \Phi x = \Phi \psi_s = \mathrm{Ts} \tag{1}$$

where $\mathrm{T} \in R^{M \times N}$ represents the sensing matrix. Through compression perception, the N-dimensional original sparse signal s is projected to M dimensions, thus achieving the effect of dimensionality reduction and compression. Since $M < N$, the above equation cannot be directly solved for x

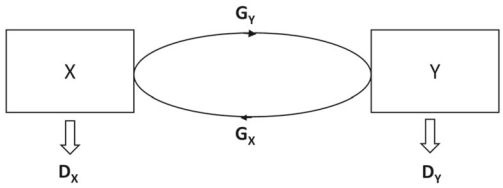


Fig. 1 Schematic diagram of cycle generative adversarial network

from y. To ensure that the original signal can be accurately reconstructed, the measurement matrix Φ is required to satisfy the restricted isometry property (RIP) [20], and x can be reconstructed to be recovered [21]. Thus, the problem of obtaining x by solving $l1$ minimum parametric, y reconstruction recovery can be formulated as:

$$\min\|s\|_1 \quad s.t.y = \mathrm{Ts} \tag{2}$$

In addition, it can also be used by matching pursuit (MP) algorithm [22], stagewise orthogonal matching pursuit (StOMP) [23], compressive sampling matching pursuit (CoSaMP) [24], and other algorithms to reconstruct the original signal x.

2.2 Generative adversarial network

Since the generative adversarial network was first presented by Goodfellow et al. [25] in 2014, it has had a growing number of profound applications in computer vision [26], natural language processing [27], human–computer interaction [28], and other domains. Among them, GAN has achieved great success in image generation, producing many popular architectures, such as DCGAN [29], StyleGAN [30], and StackGAN [31].

Cycle generative adversarial network (Cycle-GAN) [32] is a variant of the generative adversarial network, which consists of two generators and two discriminators. The generators generate false samples that are real enough to deceive the discriminators as much as possible; the discriminators distinguish between the false samples generated by the generators and the real samples as much as possible, and the two continuously confront each other and adjust their parameters to finally reach the Nash equilibrium point, i.e., the discriminators can no longer distinguish whether the samples generated by the generators are real or not. The model is shown in Fig. 1.

Among them, D_X and D_Y are the discriminators of X-domain and Y-domain respectively, which are used to discriminate whether the generated images belong to the domain; G_Y denotes the generator that generates Y-domain images from X-domain images; G_X denotes the generator that generates X-domain images from Y-domain images. The objective of Cycle-GAN is to learn the mapping from X to Y and from Y to X. Cycle-GAN is a technique that automatically performs image-to-image transformations without the need for pairwise examples. In addition, the visual image encryption process can also be seen as a migration of different domains; therefore, we propose a learning visual image encryption scheme based on Cycle-GAN to realize encryption from the plaintext domain to the ciphertext domain.

2.3 Algorithm descriptions

The overall flow of visual image encryption is shown in Fig. 2, the blue box on the left shows the compression sense and chaotic encryption process, while the right part shows the visual encryption and decryption process. First, the plaintext image is sparse by discrete wavelet transform (DWT) and compressed with CS; then, the ciphertext image is obtained by permutation and diffusion through the improved Henon map. Finally, feeding the ciphertext image into the encryption network causes it to migrate from the ciphertext domain to the plaintext domain, generating a completely uncorrelated visual image. As shown in the figure, the generated visual images are stored in the database, and the receiver can find and decrypt the visual images from the database by using the neural network and the key, where the key is the trained network parameters.

2.4 Network architecture

The neural network architecture of the encryption network and decryption network in this paper is based on Cycle-GAN, as seen in Fig. 3. The encryption network gradually extracts the image features through three convolutional layers and generates the target image through three deconvolutional layers to simulate the encoding and decoding process of the image. For each convolutional layer, an InstanceNormalization layer and the ReLU excitation function are adopted to perform normalization operation and activation. Finally, the Tanh function is employed to map to [− 1 1]. In addition, 18 residual blocks [33] are added between the encoding and decoding parts to ensure that the input data information from the previous network layer is directly applied to the later network layers, so that the deviation of the corresponding output from the original input is reduced; otherwise, the features of the original image will not be retained in the output and the output will deviate from the target contour. The network structure of the encryption network is shown in Table 1, and the decryption network is the same.

2.5 Training process

As shown in Fig. 3, the encryption network G_{XY} is used to encrypt the original input images, the decryption network

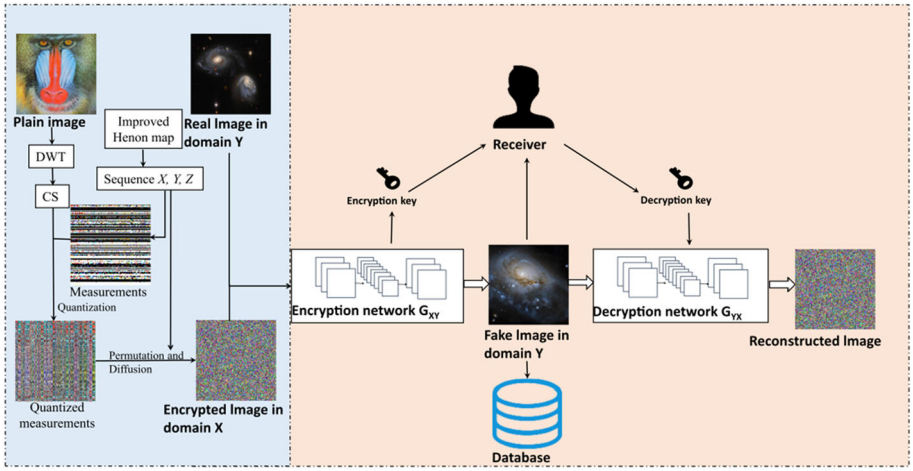


Fig. 2 The process of the visual encryption algorithm

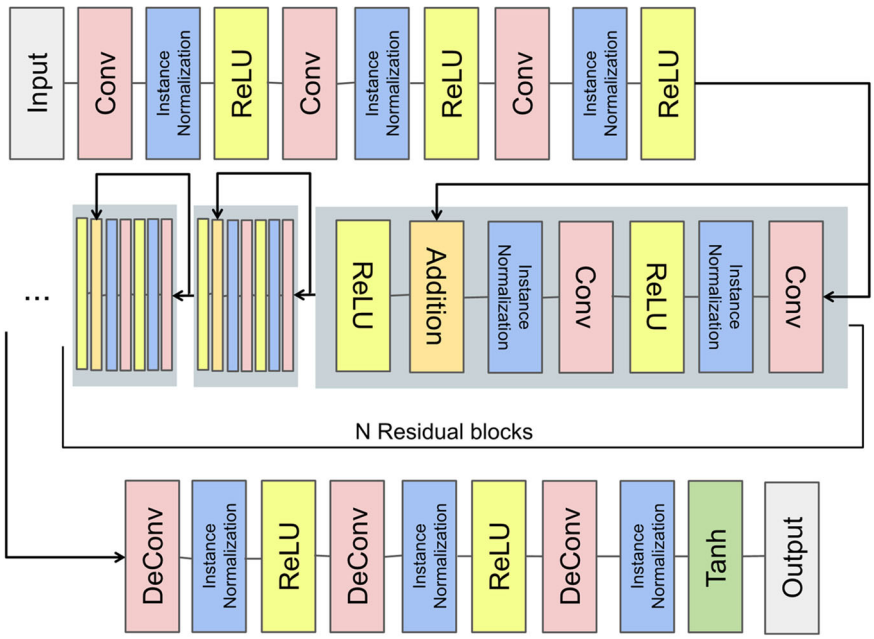


Fig. 3 Network architecture

Table 1 Neural network structure

Layer name	Number	Kernel size	Input channels	Output channels	Parameters	Total parameters
Convolution1	1	7 × 7	6	64	18,880	18,880
Convolution2	1	3 × 3	64	128	73,856	92,736
Convolution3	1	3 × 3	128	256	295,168	387,904
Residual blocks	18	3 × 3	256	256	590,080	11,009,344
Deconvolution1	1	3 × 3	256	128	295,040	11,304,384
Deconvolution2	1	3 × 3	128	64	73,792	11,378,176
Deconvolution3	1	7 × 7	64	3	9411	11,387,587

G_{YX} is responsible for recovering the image, and the discriminator network D is mainly used to distinguish whether the generated image is the same as the target domain or not. The overall loss L used in the model training process is as follows:

$$L = L_{\mathrm{E}} + L_{\mathrm{D}} + \lambda L_R + \mu L_{\mathrm{F}} \tag{3}$$

where the L_{E} indicates the loss of the encryption network, L_{D} indicates the loss of the discriminator network, L_R indicates the loss of the decryption network, and L_{F} indicates the loss of the feature, it is used to acquire as many features of the target image as possible, constraining features of the input image to transform arbitrarily, thus improving the quality of the generated image. Specifically, the loss $\mathrm{L_E}$ of the encrypted network is:

$$\mathcal{L}_{\mathrm{E}} = E_{x\sim p_{\mathrm{data}}(x)}\big[\log(1 - D_Y(G(x))\big] \tag{4}$$

where G represents an encryption network, and D represents the discriminator network. The loss of G is to minimize the success rate of the discriminator network D in discriminating the image generated by the encryption network G, while the loss LD of the discriminator network D can be seen as maximizing the classification accuracy, which is the exact opposite of the goal of the encryption network, and is formulated as follows:

$$L_D = E_{x\sim \mathrm{pdata}(x)}\log D(x) + E_{x\sim \mathrm{pdata}(x)}\log(1 - D(G(x))) \tag{5}$$

When the two networks, the encryption network y and the discriminator network, reach a balanced state, the discriminator network D will achieve 50% classification accuracy for the generated images, that is, the generated images are very similar to the real images and cannot be distinguished from each other. In addition, in order to recover the ciphertext image with less distortion, the reconstruction loss L is defined as:

$$\begin{aligned} L_{\mathrm{R}} &= E_{x\sim p_{\mathrm{data}(x)}}\|Y - X\|_2 = E_{x\sim p_{\mathrm{data}(x)}}\sum_{i=1}^{n}|y_i - x_i| \\ &= E_{x\sim p_{\mathrm{data}(x)}}(|y_1 - x_1| + \cdots + |y_i - x_i|) \end{aligned} \tag{6}$$

The image completes domain-specific transformations while also causing random changes in the irrelevant domains. Therefore, it is necessary to constrain and guide the image generation process by introducing a feature loss function in order to ensure the feature transformation of the transformed domain while preserving the features of the irrelevant domain as much as possible. It is defined as follows:

$$\mathcal{L}_{\mathrm{F}}(G,\ F) = E_{x-P_{\mathrm{dater}}}\big[\|F(x) - x\|_2\big] \tag{7}$$

Before the model training starts, the convolution layer first requires a random initialization of the parameters:

$$W_n = \mathrm{random}\big[w_{n,1},\ w_{n,2},\ \ldots,\ w_{n,j}\big] \tag{8}$$

where W_n represents the parameters of the convolutional layer and $W_{n,j}$ is the$j$th parameter of the nth convolutional layer.

When the network is trained, the loss function is computed to measure the difference between the predicted outcome and the target and transmitted back to the entire neural network, thus guiding the network to progressively update the parameters to reduce the error. It is actually a gradient descent, and the formula is as follows:

$$\begin{aligned} \theta_j &= \theta_j - \alpha \vee J(\theta_j) = \theta_j - \alpha\frac{\delta}{\theta_j}J(\theta_j) \\ &= \theta_j - \frac{1}{m}\alpha\frac{\delta}{\theta_j}\sum_{i=1}^{m}\big[(E(x_i + y_i) - y_i)^2 + (F(E(x_i + y_i)) - x_i)^2\big] \end{aligned} \tag{9}$$

where θ_j is the value of parameter θ in the$j$th training epoch, α is the learning rate, and $\vee J(\theta_j)$ is the gradient that passed back to θ_j.

2.6 Encryption and decryption process

The detailed steps for visual image encryption are as follows:

Step 1: Resize an image of size $m \times n$ into a 1D binary matrix and obtain the sparsification matrix N by DWT.

Step 2: Set the initial value a_1, b_1, r_1, x_1, and y_1 of the improved Henon map, then iterate it for $m*n$ times and generate a chaotic sequence $X_i = \{X_1, X_2, \ldots, X_i\}$.i = 1,2,…,$m \times n$, the improved Henon map which can be expressed as Eq. (10) [34]:

$$\begin{cases} x_{n+1} = \left(1 - ax_n^2 + y_n\right)\sin\left(\frac{r\pi}{x_n}\right) \\ y_{n+1} = bx_n\sin\left(\frac{\pi}{y_n}\right) \end{cases} \tag{10}$$

where a, b, and r represent parameters,x_n and y_n mean the state values. The motion trajectory of improved Henon map is distributed in a bigger range and has better chaotic performances.

Step 3: Set the image signal length$L$ and sampling rate R, the measurement value M is obtained by multiplying the signal length by the sampling rate, followed by the equation to obtain the measurement matrix:

$$\phi_{\mathrm{A}} \leftarrow \frac{1}{N}\mathrm{randn}(\mathbf{x}_i,\ M,\ N) \tag{11}$$

where Φ_A is the measurement matrix, N and M represent the number of input samples and measured samples.

Step 4: The measurement matrix is then quantified by the following equation to achieve a sequence J:

$$z = \text{round}\left(a_1 \cdot \left(1 + e^{-a_2(y-a_3)}\right)^{-1}\right) \tag{12}$$

where round(·) denotes rounding the elements to the nearest integer, $a_1 = 255$, $a_2 = \Phi_{Amax} - \Phi_{Amin}$, $a_3 = (\Phi_{Amax} + \Phi_{Amin})/2$, where Φ_{Amax} and Φ_{Amin} are the maximum and minimum of the measurements. After quantization, the value of measurements is an integer between 0 and 255.

Step 5: Set the initial value a_2, b_2, r_2, x_2, and y_2 of the improved Henon map, then iterate it for $m \times n$ times and generate a chaotic sequence $Y = \{Y_1, Y_2, \ldots, Y_i\}$.i = 1,2,…,$m \times n$;

Step 6: The resulting chaotic sequence Y is sorted from smallest to largest to form an ordered sequence Y' and the number of each value in the sequence is determined in the sequence Y to form a replacement address set S. The sequence J is replaced by the replacement address set S to obtain C;

Step 7: Set the initial value a_3, b_3, r_3, x_3, and y_3 of the improved Henon map to generate a chaotic sequence$Z = \{Z_1, Z_2, \ldots, Z_i\}$ by iterating it for $m \times n$ times;

Step 8: Let the scrambled sequence C and chaotic sequence Z perform binary XOR operation, then the encrypted image is obtained.

After obtaining the ciphertext image, we convert it into a digital matrix and map the pixel values to the range [− 1, 1], then we put it into the cryptographic network, traverse and normalize the image matrix, and activate it by the convolution kernel in the convolution layer to form the input feature matrix for the next convolution layer. After going through all the convolutional layers, new predictions are obtained to combine into a new matrix. Finally, we output the matrix of the last convolutional layer and convert it to uint8 format to implement the visual image.

The overall flow of the decryption process is shown in Fig. 4. The orange part on the left side shows the visual decryption process, and the blue part on the right side shows the chaos decryption and compression-aware reconstruction process. First, the receiver inputs the visual image and the key into the decryption network to obtain the ciphertext image; then, the quantized image is obtained by inverse permutation and diffusion operation of the ciphertext image through chaos mapping. Finally, the plaintext image is obtained by inverse quantization, omp reconstruction algorithm, and inverse wavelet transform.

3 Simulation results and security analysis

3.1 Simulation results

In our experiment, the chaotic system with the initial values $a_1 = 1, b_1 = 0.3, r_1 = 0.1, x_1 = y_1 = 0.5, a_2 = 1, b_2 = 0.3, r_2 = 0.1, x_2 = y_2 = 0.6, a_3 = 1, b_3 = 0.3, r_3 = 0.1$, and $x_3 = y_3 = 0.7$ is set. For the deep neural network, we chose Adam solver as the parameter optimizer with a batch size of 1 and an initial learning rate of 0.0002. All the experiments are carried out on 64-bit Windows 11, and the hardware includes an A100-PCIE-40GB GPU. To evaluate the encryption method, the source image we chose was a color image from the NASA Hubble Space Telescope images. The size of the training set is 1000, its resolution is 256 × 256, the training epoch is 600, and the test set is 50.

The visualized image encryption process is shown in Fig. 5. Firstly, the secret image is encrypted into a ciphertext image by compressed perception and chaotic system; then, further visualized encryption is performed by neural network, as seen from the figure, the generated image is like a real space image, no information about the secret image can be seen, and the reconstructed ciphertext image is not different from the encrypted image. In addition, the decrypted image is the same as the secret image, indicating that the ciphertext image can be decrypted successfully with high image quality.

In addition, we tested the weights of reconstruction loss and feature loss and plotted the loss curves for the loss values. In these graphs, the green line depicts the variation of reconstruction loss values with epoch and the brown line depicts the variation of the feature loss values. Figure 6 shows the trend of loss values for different choices of reconstruction loss hyperparameters. It can be seen that by setting $\lambda = 1$ and $\lambda = 3$, the loss values do not decrease m as fast as $\lambda = 2$ and eventually do not smooth out, while $\lambda = 2$ has a loss value very close to 0 after 600 iterations. In additional, Fig. 7 shows the trend of the loss values for different hyperparameters of the feature loss. It can be seen that the final values obtained for the loss values for $\mu = 1$ and $\mu = 10$ are not as small as those for $\mu = 50$, i.e., the least feature loss. It illustrates that when the weights of the loss function are not set appropriately, it can greatly lead to the appearance of outliers during the training process, making the training effect poor. And when outliers occasionally appear as in Fig. 4 (b), they may be caused by outliers or noise in the dataset, but have no effect on the final training effect. Therefore, by setting $\lambda = 2$ and $\mu = 50$, our model can obtain the best image reconstruction quality.

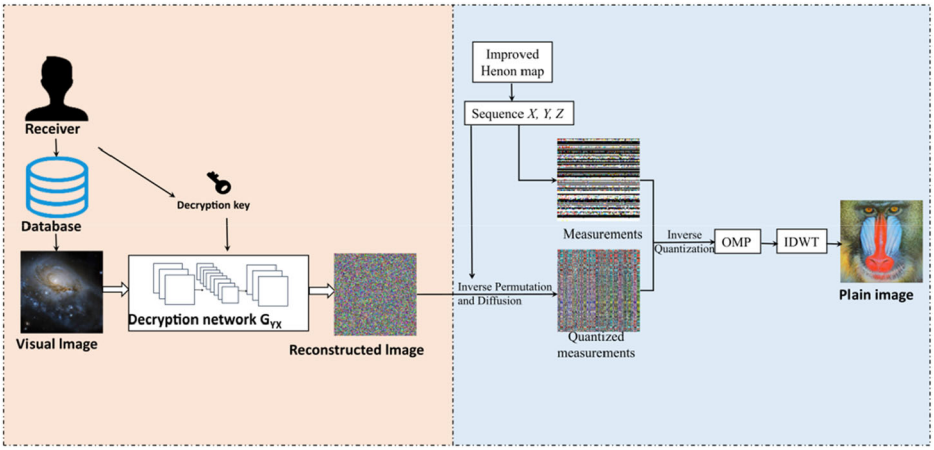


Fig. 4 The decrypted process of the visual encryption algorithm

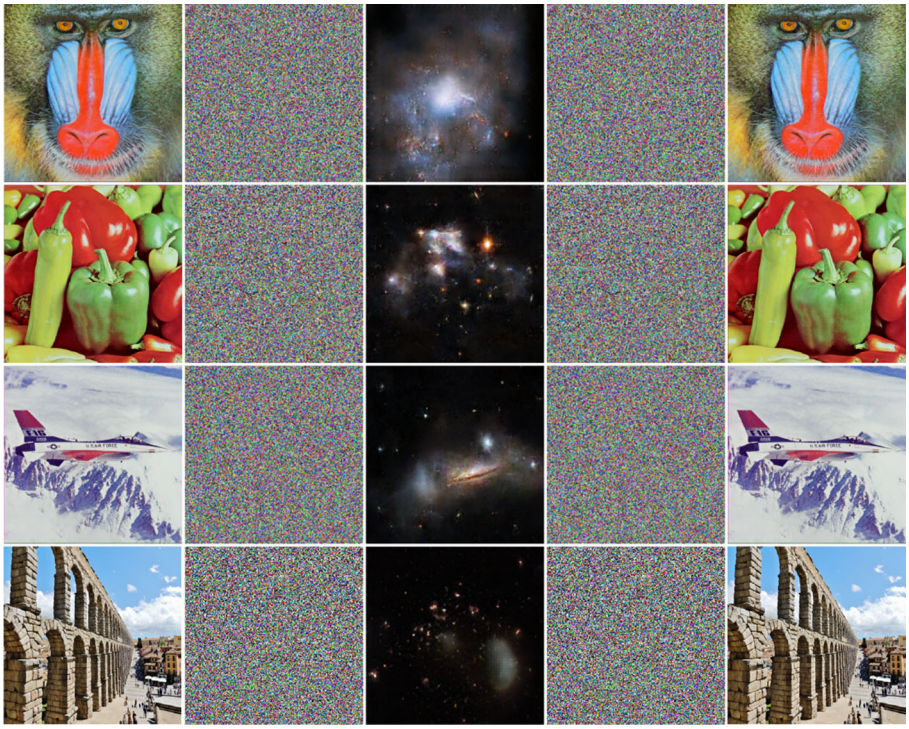


Fig. 5 Visual encrypt process and decryption process. From left to right: secret image, encrypted image, visual image, recovered image, and decrypted image

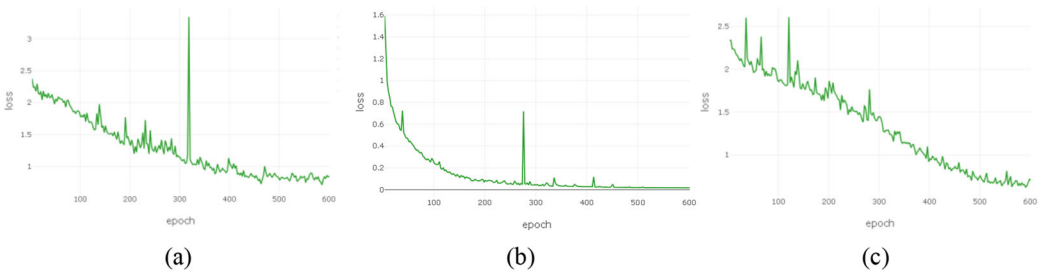
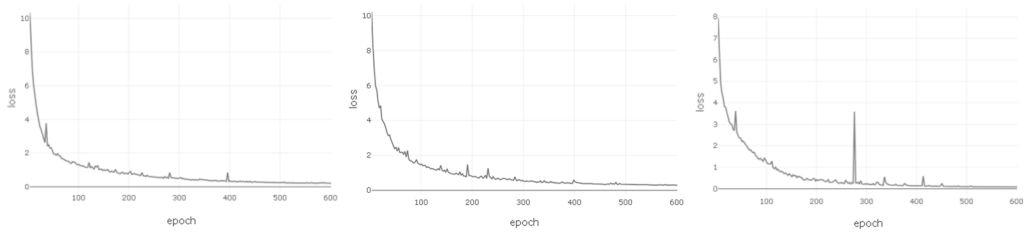


Fig. 6 Reconstruction loss of different hyperparameters. From left to right: $\lambda = 1$, $\lambda = 2$, $\lambda = 3$



(a) (b) (c)

Fig. 7 Feature loss during different hyperparameters. From left to right: $\mu = 1$, $\mu = 10$, $\mu = 50$

3.2 Key security analysis

3.2.1 Security key space

The key space refers to the set of keys that can be used for encryption and is an important feature of a cryptosystem. *n* our scheme, the initial values of the chaotic mapping and the training parameters of the neural network form the key set. When the precision of each floating point number is 10^{-14}, the key space will be at least $(10^{210})^{11387587}$. Therefore, the key space is over 2^{100} and can resist exhaustive attacks [35].

3.2.2 Key sensitivity analysis

Due to the complexity of deep neural networks, its error will propagate between the layers of the model. During the convolution process, the feature mapping of the current layer will be transferred to the next layer through the convolution kernel. As the depth of the convolutional network increases, the error in the feature points will have a chain reaction that will affect the subsequent series of processes. This means that even a small change in the parameters will not restore the original mapping correctly, let alone the number of parameters in the network is $(10^{210})^{11387587}$. Therefore, the high uncertainty in the initialization and training of neural network parameters can make the model parameters different for each training, which is equivalent to one density at a time.

3.3 Ciphertext security analysis

3.3.1 Entropy analysis

Entropy represents the uniformity of information distribution and is a basic measure of cryptographic systems. It can be defined as Eq. (13) [36]:

$$H(x) = -\sum_{i=1}^{L} P(x_i)\log_2 P(x_i) \tag{13}$$

where $P(x_i)$ is the probability of x_i and N is the total number of x_i. Table 2 lists the information entropy of the encrypted images by different methods. It can be seen that our entropy value is closer to 8, which is better than the other methods. This means that our algorithm has a good encryption effect, indicating that the information distribution is uniform and attackers cannot get valid information from it.

Table 2 The entropy of the test images

Method	Entropy
Original image	0.7132
VMEI [9]	0.7846
STP-CS [11]	0.8621
Lorenz and CS [12]	0.9898
Cycle-GAN with diffusion [17]	0.9972
Ours	0.9994

3.3.2 Histogram analysis

Histograms visually reflect the distribution of pixel values, and attackers can use them to analyze the statistical rules to decipher an encryption scheme. Figure 8 shows the three-channel histograms of a set of color images. The distribution of pixel values in the original image is steep, while the encrypted image is average. Obviously, our scheme can effectively disrupt the original distribution of pixels to resist statistical analysis.

3.3.3 Correlation analysis

In digital images, neighboring pixels are often highly correlated with each other, and attackers can thus recover the original image. Therefore, pixel correlation is an important metric of an encryption algorithm. It is obtained from the following Eqs. (14)–(17) [37]:

$$R_{xy} = \frac{\cos(x,\ y)}{\sqrt{D(x)\sqrt{D(y)}}} \tag{14}$$

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{15}$$

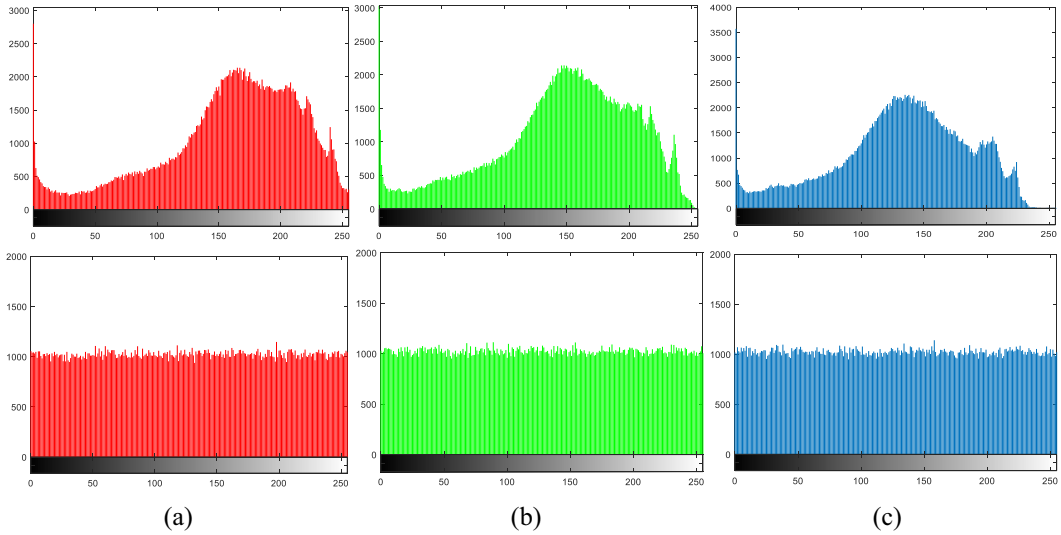


Fig. 8 Histogram analysis. The two rows are the histogram of the original image and the corresponding cipher image

Table 3 Correlation coefficients values

Method	Encrypted image		
	Horizontal	Vertical	Diagonal
Original image	0.9682	0.9724	0.9836
VMEI [9]	0.0391	0.0216	0.0164
STP-CS [11]	0.0144	0.0435	0.0301
Lorenz and CS [12]	0.0284	0.0569	0.0060
Ours	0.0059	− 0.0033	0.0061

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2 \tag{16}$$

$$\mathrm{cov}(x,\ y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)) \tag{17}$$

where x, y are the adjacent pixels, R_{xy} is the correlation between two adjacent pixels, N is the chosen number of pixels in $M \times N$ images, cov (x,y) is the covariance of two pixels: x and y.$D(x)$ is the variance, and $E(x)$ is the mean.

In the experiments, we calculate the correlation by randomly selecting 5000 pairs of adjacent pixels from the vertical, horizontal, and diagonal directions of plaintext and the corresponding ciphertext image. The results for the original image RGB three channels are shown in Fig. 9, where most of the pixel pairs are distributed along the diagonal direction. However, the pixel dots fill the whole plane in Fig. 10, which fully surfaces the very low correlation of the encrypted image. Therefore, our scheme is able to disrupt pixel correlation effectively.

In addition, the pixel correlation data are given in Table 3.

Table 4 PSNR values of different methods

Method	PSNR		
Compression ratio	0.25	0.5	0.75
STP-CS [11]	32.14	32.31	32.39
Lorenz and CS [12]	31.77	31.89	31.96
Ours	35.10	35.32	35.58

After the encryption operation, the pixel correlation coefficients in the three directions change from close to 1 to close to 0 and the correlation coefficient of adjacent pixels of our encrypted image is the lowest compared with other methods. Therefore, the encryption scheme will be effective in resisting statistical attacks.

3.4 Decryption quality analysis

3.4.1 Compression capacity analysis

To further evaluate the compression performance of this scheme, we performed a decryption operation on the ciphertext image and reconstructed the secret image using the OMP algorithm, and calculated the PSNR values at different compression ratios, and the data are shown in Table 4. It can be seen that our scheme obtains better quality under different compression ratios compared with other schemes.

3.4.2 Visual quality comparison

To evaluate the quality of the scheme reconstructed images, peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) are used as evaluation metrics. MSE and PSNR

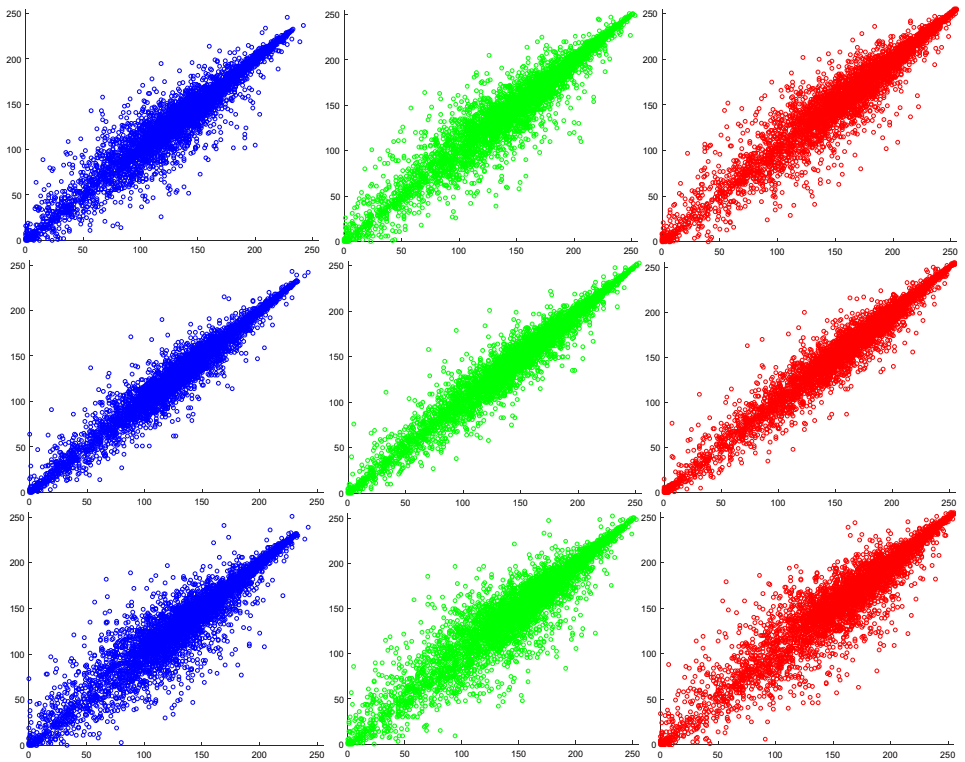


Fig. 9 Correlation of the original image: From top to bottom rows are the horizontal, the vertical, and the diagonal pixel correlations, the color represents the three channels of the RGB image

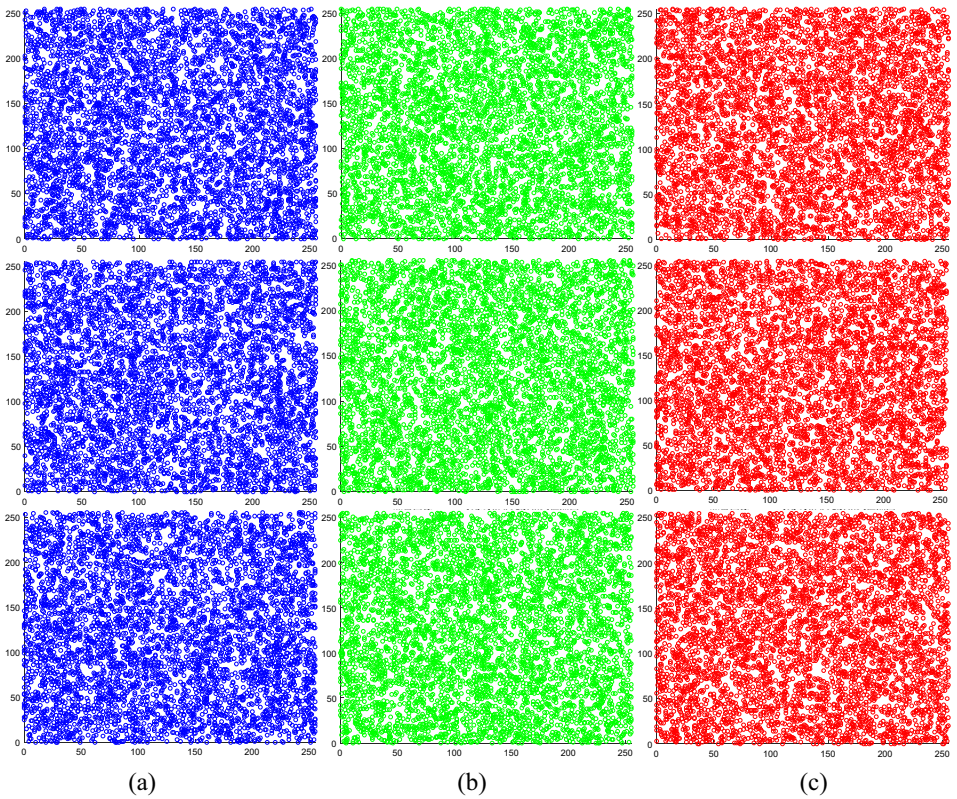


Fig. 10 Correlation of the encrypted image: From top to bottom rows are the horizontal, the vertical pixel, and the diagonal pixel correlations

Table 5 Quantitative Evaluation Results of Different Methods

Method	Metric	
	PSNR	SSIM
VMEI [9]	32.1374	0.854
STP-CS [11]	33.2678	0.8294
Lorenz and CS [12]	33.8625	0.9948
Cycle-GAN with diffusion [17]	33.1800	0.9360
DeepEDN [33]	35.8952	0.9124
Ours	52.5432	0.9958

are calculated using the following Eqs. (18)–(20) [38, 39]:

$$\mathrm{MSE} = \frac{1}{M*N}\sum_{i=1}^{M}\sum_{j=1}^{N}(X(i,\ j) - Y(i,\ j))^2 \tag{18}$$

$$\mathrm{PSNR} = 10\log_{10}\frac{(2^n-1)^2}{\mathrm{MSE}} \tag{19}$$

where M, N is the size of the image, $X(i,j)$ and $Y(i,j)$ are the pixel values of the image locations, and n is the number of pixel bits. A larger PSNR value indicates less distortion. The formula for SSIM is as follows:

$$\mathrm{SSIM}(x,\ y) = [l(x,\ y)]^{\alpha}[c(x,\ y)]^{\beta}[s(x,\ y)]^{\gamma} \tag{20}$$

where a, B, and Y represent the percentage of different features in the SSIM measure, respectively, $l(x,\ y)$ is the brightness comparison, $c(x,\ y)$ is the contrast comparison, and $s(x,\ y)$ is the structure comparison. The closer the SSIM is to 1, the more similar the two images are. The evaluation metrics of the reconstructed images according to different methods are shown in Table 5. As seen from the data, the SSIM of our method is closer to 1, i.e., the image reconstructed by our scheme is of higher quality and more similar to the original image, and outperforms the existing encryption methods.

3.5 Robust analysis

3.5.1 Differential attack analysis

A differential attack means that an attacker encrypts the original image with a few modifications and compares it with the original ciphertext image to find the relationship between them and to crack it. Therefore, two important variables are used to measure the difference between the two images: The Number Pixel Changing Rate (NPCR) and Unified Average Changing Intensity (UACI). NPCR represents the number of changed pixels and UACI delegates the average change intensity. Their corresponding ideal values are 99.6094% and 33.4635%. Their formulas use the following Eqs. (21)–(23) [40]:

Table 6 Differential attack analysis

Method	NPCR	UACI
VMEI [9]	0.9956	0.3186
Cycle-GAN with diffusion [17]	0.9964	0.3350
DeepEDN [33]	0.9963	0.3348
DeepKeyGen [41]	0.9958	0.2341
Ours	0.9962	0.3344

$$NPCR = \frac{\Sigma_{i,j}D(i,\ j)}{M*N} * 100\% \tag{21}$$

$$D(i,\ j) = f(x) = \begin{cases} 1, & C_1(i,\ j) \neq C_2(i,\ j) \\ 0, & \text{otherwise} \end{cases} \tag{22}$$

$$\mathrm{UACI} = \frac{1}{M*N}\frac{\sum(C_1(i,\ j) - C_2(i,\ j))}{255} * 100\% \tag{23}$$

where $M * N$ are the size of an image, C_1 $(i.\ j)$ and C_2 $(i.\ j)$ are the pixel difference, and D $(i.\ j)$ is used to determine whether they are the same.

To perform differential attack analysis, we change a pixel value randomly in a secret image and then encrypt it to generate a new cryptographic image. Table 6 lists the evaluation indicator data, it can be seen that the evaluation index of our scheme is closer to the ideal value than other algorithms. Which evinces that our scheme has high plaintext sensitivity to resist differential attacks.

3.5.2 Noise attack analysis

During Internet communication, it is very likely that images will be partially lost, distorted, and contaminated, and these effects are very common. Therefore, a feasible encryption scheme should be robust enough to properly handle lossy images.

To evaluate the robustness of the scheme, we added 0.01%, 0.1%, 1%, and 3% salt and pepper noise in the encrypted image and then decrypted them. The results are shown in Fig. 11, indicating that our scheme can still recover the original image well even when the data loss rate reaches 3%, the scheme is robust and can keep the decrypted medical image clear.

3.6 Time efficiency analysis

Algorithm efficiency is an important indicator of encryption performance and application value; to demonstrate the efficiency of our proposed scheme, images of size 256 × 256

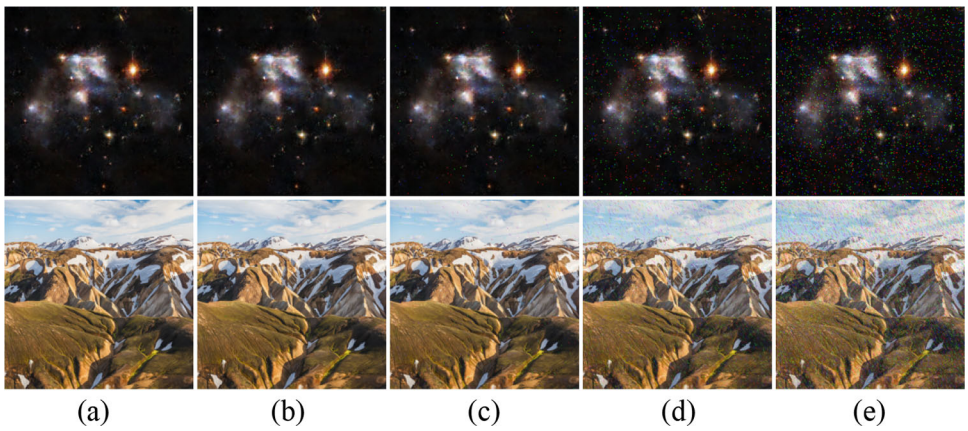


Fig. 11 Noise attacks: The top to bottom rows are the ciphertext images and their corresponding decryption results **a** original ciphertext image; **b** 0.01% salt and pepper noise; **c** 0.1% salt and peppers noise; **d** 1% salt and peppers noise **e** 3% salt and peppers noise

Table 7 Time efficiency analysis (Unit: s)

Method	Encryption time	Decryption time	Total time
STP-CS [11]	0.7362	0.6729	1.4091
Lorenz and CS [12]	5.7214	5.2193	10.9407
DeepEDN [33]	0.1911	0.1631	0.3542
Ours	0.1733	0.1264	0.2997

are tested and Table 7 lists the encryption, decryption, and total time spent. From the data, it can be seen that the time efficiency of our scheme is higher than similar state-of-the-art methods, which has better performance and application value.

4 Conclusions

This article introduces a learning visual image encryption scheme based on Cycle-GAN and compressed sensing, which focuses on protecting content security while also ensuring visual security. We add feature loss to ensure the image level correspondence between image domains, which better preserves the background color of the input image and irrelevant features. We also test the hyperparameters of the decryption loss λ and the hyperparameter of feature loss μ to train the model with the best performance. The experiment results demonstrate that our scheme possesses a large key space and strong key sensitivity. Moreover, compared with visual encryption schemes that embed images into a cover image, our solutions help to improve the sharpness of the generated images and are resistant to steganalysis.

Funding This project is supported in part by the National Natural Science Foundation of China: 62262062, the major programs incubation plan of Xizang Minzu University: 22MDZ03, and the Research Team Project for Xizang-related Network Information Content and Data Security (No. 324042000709).

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

Declarations

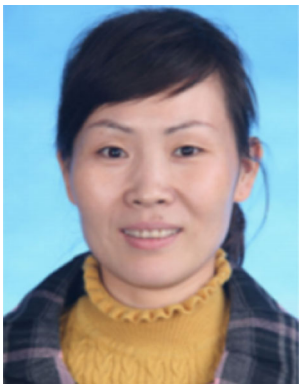
Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Kesavan, K. K., Kumar, M. R.: Optical color image encryption based on Hartley transform and double random phase encoding system. In: 2011 3rd International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), pp. 1–3. IEEE (2011)
2. Kumar, M. R., Linslal, C. L., Pillai, V. M., Krishna, S. S.: Color image encryption and decryption based on jigsaw transform employed at the input plane of a double random phase encoding system. In: International Congress on Ultra Modern Telecommunications and Control Systems, pp. 860–862. IEEE (2010)
3. Kesavan, K. K., Kumar, M. R.: Optical data security using Hartley transform, pixel scrambling and chaos theory for images. In: 2011 3rd International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), pp. 1–5. IEEE (2011)
4. Bezerra, J.I.M., Machado, G., Molter, A., Soares, R.I., Camargo, V.: A novel simultaneous permutation–diffusion image encryption scheme based on a discrete space map. Chaos Solitons Fractals **168**, 113160 (2023)
5. Xian, Y., Wang, X., Zhang, Y., Yan, X., Leng, Z.: A novel chaotic image encryption with FSV based global bit-level chaotic permutation. Multimed. Tools Appl. **82**(1), 407–426 (2023)
6. Xu, S., Wang, X., Ye, X.: A new fractional-order chaos system of Hopfield neural network and its application in image encryption. Chaos Solitons Fractals **157**, 111889 (2022)
7. Yan, S., Gu, Z., Park, J.H., Xie, X.: Synchronization of delayed fuzzy neural networks with probabilistic communication delay and its application to image encryption. IEEE Trans. Fuzzy Syst. **31**(3), 930–940 (2022)
8. Shamsi, Z., Laiphrakpam, D. S.: Securing encrypted image information in audio data. Multimed. Tools Appl. 1–23 (2023)

9. Bao, L., Zhou, Y.: Image encryption: generating visually meaningful encrypted images. Inf. Sci. **324**, 197–207 (2015)
10. Ye, G., Pan, C., Dong, Y., Jiao, K., Huang, X.: A novel multi-image visually meaningful encryption algorithm based on compressive sensing and Schur decomposition. Trans. Emerg. Telecommun. Technol. **32**(2), e4071 (2021)
11. Ping, P., Yang, X., Zhang, X., Mao, Y., Khalid, H.: Generating visually secure encrypted images by partial block pairing-substitution and semi-tensor product compressed sensing. Digit. Signal Process. **120**, 103263 (2022)
12. Ren, H., Niu, S., Chen, J., Li, M., Yue, Z.: A visually secure image encryption based on the fractional lorenz system and compressive sensing. Fractal Fract. **6**(6), 302 (2022)
13. Wang, C., Zhang, Y.: A novel image encryption algorithm with deep neural network. Signal Process. **196**, 108536 (2022)
14. Shen, H., Li, X., Zhang, L., Tao, D., Zeng, C.: Compressed sensing-based inpainting of aqua moderate resolution imaging spectroradiometer band 6 using adaptive spectrum-weighted sparse Bayesian dictionary learning. IEEE Trans. Geosci. Remote Sens. **52**(2), 894–906 (2013)
15. Xiuli, C., Zhihua, G., Yiran, C., Yushu, Z.: A visually secure image encryption scheme based on compressive sensing. Signal Process. **134**, 35–51 (2017)
16. Li, X., Jiang, Y., Chen, M., Li, F.: Research on iris image encryption based on deep learning. EURASIP J. Image Video Process. **2018**(1), 1–10 (2018)
17. Bao, Z., Xue, R.: Research on the avalanche effect of image encryption based on the Cycle-GAN. Appl. Opt. **60**(18), 5320–5334 (2021)
18. Panwar, K., Kukreja, S., Singh, A., Singh, K.K.: Towards deep learning for efficient image encryption. Procedia Comput. Sci. **218**, 644–650 (2023)
19. Donoho, D.L.: Compressed sensing. IEEE Trans. Inf. Theory **52**, 1289–1306 (2006)
20. Bandeira, A.S., Dobriban, E., Mixon, D.G., Sawin, W.F.: Certifying the restricted isometry property is hard. IEEE Trans. Inf. Theory **59**(6), 3448–3450 (2013)
21. Baraniuk, R.G.: Compressive sensing. IEEE Signal Process. Mag. **24**, 118–121 (2007)
22. Krstulovic, S., Gribonval, R.: MPTK: matching pursuit made tractable. In: 2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings, Vol. 3, pp. III–III. IEEE (2006)
23. Donoho, D.L., Tsaig, Y., Drori, I., Starck, J.L.: Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit. IEEE Trans. Inf. Theory **58**(2), 1094–1121 (2012)
24. Zhang, X., Xu, W., Cui, Y., Lu, L., Lin, J.: On recovery of block sparse signals via block compressive sampling matching pursuit. IEEE Access **7**, 175554–175563 (2019)
25. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Bengio, Y.: Generative adversarial networks. Commun. ACM **63**(11), 139–144 (2020)
26. Cao, Y.J., Jia, L.L., Chen, Y.X., Lin, N., Yang, C., Zhang, B., Dai, H.H.: Recent advances of generative adversarial networks in computer vision. IEEE Access **7**, 14985–15006 (2018)
27. Zhu, Y., Zhang, Y., Yang, H., Wang, F.: GANCoder: an automatic natural language-to-programming language translation approach based on GAN. In: Natural Language Processing and Chinese Computing: 8th CCF International Conference, NLPCC 2019, Dunhuang, China, October 9–14, 2019, Proceedings, Part II 8, pp. 529–539. Springer International Publishing (2019)
28. Zhuang, Q., Gan, S., Zhang, L.: Human-computer interaction based health diagnostics using ResNet34 for tongue image classification. Comput. Methods Programs Biomed. **226**, 107096 (2022)
29. Fang, W., Zhang, F., Sheng, V.S., Ding, Y.: A method for improving CNN-based image recognition using DCGAN. Comput. Mater. Contin. **57**(1), 167–178 (2018)
30. Abdal, R., Qin, Y., Wonka, P.: Image2stylegan: How to embed images into the stylegan latent space? In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4432–4441 (2019)
31. Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., Metaxas, D. N.: Stackgan: text to photo-realistic image synthesis with stacked generative adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 5907–5915 (2017)
32. Zhang, Y., Liu, S., Dong, C., Zhang, X., Yuan, Y.: Multiple cycle-in-cycle generative adversarial networks for unsupervised image super-resolution. IEEE Trans. Image Process. **29**, 1101–1112 (2019)
33. Ding, Y., Wu, G., Chen, D., Zhang, N., Gong, L., Cao, M., Qin, Z.: DeepEDN: A deep-learning-based image encryption and decryption network for internet of medical things. IEEE Internet Things J. **8**(3), 1504–1518 (2020)
34. Gao, X.: A color image encryption algorithm based on an improved Hénon map. Phys. Scr. **96**(6), 065203 (2021)
35. Farah, M.A., Farah, A., Farah, T.: An image encryption scheme based on a new hybrid chaotic map and optimized substitution box. Nonlinear Dyn. **99**(4), 3041–3064 (2020)
36. Tang, Z., Zhang, X., Lan, W.: Efficient image encryption with blockshuffling and chaotic map. Multimed. Tools Appl. **74**, 5429–5448 (2015)
37. Chen, G., Mao, Y., Chui, C.K.: A symmetric image encryption scheme based on 3D chaotic cat maps. Chaos Solitons Fractals **21**, 749–761 (2004)
38. Man, Z., Li, J., Di, X., Bai, O.: An image segmentation encryption algorithm based on hybrid chaotic system. IEEE Access **7**, 103047–103058 (2019)
39. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error measurement to structural similarity. IEEE Trans. Image Process. **13**, 600–612 (2004)
40. Belazi, A., El-Latif, A.A.A., Belghith, S.: A novel image encryption scheme based on substitution-permutation network and chaos. Signal Process. **128**, 155–170 (2016)
41. Ding, Y., Tan, F., Qin, Z., Cao, M., Choo, K.K.R., Qin, Z.: DeepKeyGen: a deep learning-based stream cipher generator for medical image encryption and decryption. IEEE Trans. Neural Netw. Learn. Syst. **33**(9), 4915–4929 (2021)

Zhaoyang Liu was born in Jiangsu, China. Currently, he is pursuing Master Degree in Xizang Minzu University, China. His research interests include image encryption, deep learning and information security.



Ru Xue was born in Shaanxi, China. She received the B.S. degree from Xi'an University of Technology, Xi'an, ShaanXi, China, in 1998, the M.S. degree from Tianjin University, Tianjin, Tianjin China, in 2007, and the Ph.D. degree from Chang'an University, Xi'an, Shaanxi, China, in 2014. Presently, she is working as a Professor in Xizang Minzu University, Xianyang, ShaanXi, China. Her research interests include computer vision, image processing, and pattern recognition.