Fu *et al. EURASIP Journal on Image and Video Processing* (2020) 2020:46
https://doi.org/10.1186/s13640-020-00534-2

EURASIP Journal on Image and Video Processing

RESEARCH Open Access

The secure steganography for hiding images via GAN

Zhangjie Fu[1,2], Fan Wang[1] and Xu Cheng[1*]

*Correspondence:
xcheng@nuist.edu.cn
[1]Department of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China
Full list of author information is available at the end of the article

Abstract

Steganography is one of the important methods in the field of information hiding, which is the technique of hiding secret data within an ordinary file or message in order to avoid the detection of steganalysis models and human eyes. In recent years, many scholars have applied various deep learning networks to the field of steganalysis to improve the accuracy of detection. The rapid improvement of the accuracy of steganalysis models has caused a huge threat to the security of steganography. In addition, another important factor that limits the security of steganography is capacity. The larger the capacity, the worse and more unnatural the visual quality of carrier images after embedded. Therefore, this paper proposes a steganography model—HIGAN, which constructs the encoding network composed of residual blocks to hide the color secret image into another color image of the same size to output a lower distortion and higher visual quality steganographic image. Moreover, it utilizes the adversarial training between the encoder-decoder network and the steganalysis model to improve the ability to resist the detection of steganalysis models based on deep learning. The experimental results show that our proposed model is achievable and effective. Compared with the previous steganography model for hiding color images based on deep learning, the steganography model in this article could achieve steganographic images with higher visual quality and stronger security.

Keywords: Steganography, Information hiding, Generative adversarial networks, Deep learning

1 Introduction

Information hiding is one of the important ways to ensure the security of information in the network, which can hide secret information into the carrier imperceptibly [1]. It can not only ensure the security of the data itself, but also ensure that the data can be transmitted securely. Steganography, as an important technology of information hiding, has gradually become one of the hot research directions of domestic and foreign scholars. In order to successfully transmit the secret information, the sender hides the secret information into the carrier in an invisible way, and the receiver extracts the secret information from the steganographic images by using the key. Images have become the main carrier of steganography because of its availability and diversity. In recent years, the detection accuracy of steganalysis models based on deep learning (such as GNCNN

[2], Xu'Net [3], Ye'Net [4], SRNet [5]) exceeds the most widely used steganalysis model—spatial rich model (SRM) [6] and its various variants [7, 8]. It poses a huge threat to the security of steganography algorithms. In addition, some steganography methods pay too much attention to the security of steganography, so they sacrifice the embedded capacity and limit it to 0.4 bit per pixel (bpp) or less in exchange for the ability of resisting the detection of steganalysis models. But, this also causes the practicality of this type of methods to be greatly reduced. Just like the work of [9], it can effectively resist the detection of steganalysis to ensure the security of steganography, but its capacity is only 0.1 bpp or even lower. Finally, the accuracy of extraction is to measure the difference between the recovered secret information and the original secret information. It directly reflects the effectiveness of a steganographic model.

In the design of traditional adaptive steganographic algorithms, researchers tried their best to find the best balance between the security and capacity. As we all know, capacity and security are related and inversely proportional. The larger the capacity, the lower the stegangraphy models and the worser the visual quality of steganographic images. Therefore, some researchers spent a lot of time and effort to obtain the smallest embedding distortion cost by calculating the embedding change probability in published works, so that the steganographic algorithms still could ensure the security under the maximum allowable embedding capacity 0.4 bpp (such as S-UNIWARD [10], WOW [11], HILL [12]). With the development of deep learning, some researchers tried to automatically find the smallest embedding distortion costs by adversarial training with steganalysis models [13–15]. However, some researchers tried to embed more binary bitstream through the encoder network, for example, steganoGAN [16]. Its maximum capacity can be increased to 4.4 bpp and the performance of security even better than the adaptive steganography algorithms by adding the critic network. Unlike traditional steganography algorithms that require extra keys to extracting secret information, the key of the steganography method based on encoder-decoder network is the parameters and weights of the trained model. Therefore, the parameters need to be shared privately to the receiver before sending the secret information. In addition, even if the model structure is shared publicly, the third party still cannot extract the secret image due to different parameter settings. But, due to the pooling and normalization operations in the encoder-decoder network, the secret information cannot be completely extracted by decoder network. In addition to embedding binary bitstreams in the carrier image, some researchers have tried to embed other types of data into the carrier images by encoder-decoder network (such as Deep Steganography [17], Rehman [18]). Deep Steganography embedded a color image into another color image with the same size, but it revealed secret images and steganographic images had a bad visual quality and some peak noise point existed. Rehman embedded a grayscale image into a color image with same size and outputed a steganographic image with the color deviation problem. However, the above steganography models used to hide an image had a special advantage, that is, even if the embedded secret image cannot be fully recovered, it will not affect the extractor's understanding of the semantic information contained in the secret images. Of course, its shortcoming is also obvious, because the carrier image is embedded with a large amount of secret information, which leads to poor steganographic security. Both in terms of the visual quality of steganographic images and the ability to resist steganalysis and detection, the performance of the above models was not very good. In order to improve these shortcomings, the works of [19]

proposed an encoding network with a similar structure to U-Net and a decoder network with fully convolutional layers to improve the visual quality of the reconstructed color secret image and steganographic color image. And the works of [20] constructed an encoder network composed of Inception V3 blocks to hide a grayscale image in the Y channel of a color carrier image to avoid modifying the Cb and Cr channels containing the brightness and color information. Moreover, it proved that even in the case of a large steganographic capacity (for example, a gray image of secret information or 4.4 bpp binary bit information), the security can still be effectively improved through adversarial training. Another thing to mention is that, this method could reduce the color deviation of the steganographic images only when the secret image is a single-channel grayscale image. Moreover, it is important to understand that the security of this type of methods cannot be compared with steganography methods that used to hide secret binary bitstream information, because the amount of information embedded in the carrier image is not the same level. Inspired by ISGAN [20], we propose a steganographic model called HIGAN. It can hide a three-channel color image other than just a one-channel grayscale image into another color image with the same size. And the low color distortion and high visual quality steganographic images are generated by the encoder network, which is similar to the structure of resnet. Compared to the works of [19], the encoder network of our model can not only fuse the feature maps of different layers, but also avoid the disappearance of the gradient during the adversarial training process. Another highlight is that our model takes into account the poor security of the existing steganographic network used to hide color images. It could improve the ability to resist the detection of steganalysis model by adversarial training between the encoder-decoder network (generator) and steganalysis model (discriminator). The contributions of our work are as follow:

1) This paper proposes the steganography model—HIGAN, which could hide a three-channel color image into another three-channel color image. In the case of large steganographic capacity, it considers the visual quality and security of steganographic images at the same time.
2) We introduce the encoder network composed of the convolutional layers and residual blocks to output the low distortion steganographic images that are more closer to the original carrier images and reduce the disappearance of gradient during the training process.
3) The experiment results show that our model is practical and effective. Compared with the previous steganography models that were use to hide the color secret images, our model adds the steganalysis model (Xu'Net) as discriminator to improve the ability of resisting the detection of steganalysis model by adversarial training with the encoder-decoder network.

The rest of the paper is organized as follows. In Section 2, we discuss the previous work of steganography models based on deep learning. In Section 3, we introduce the proposed steganography method called HIGAN in detail. In Section 4, we describe the implementation process and details of experiments, the dataset, the parameter settings, and the evaluation metrics. In Section 5, we discuss and analyze the results of experiments. Finally, the conclusion is described in Section 6.

2 Related works

In recent years, deep learning networks have been used widely in the field of information hiding. Many steganography models and steganalysis models based on deep learning have emerged. Deep learning networks can not only improve the security and capacity of steganography models, but also improve the detection accuracy of steganalysis models.

2.1 Steganography based on deep learning

In 2014, Goodfellow et al. [21] proposed generative adversarial network (GAN), which provides an opportunity for the combination of deep learning and information hiding. The existing steganography models based on deep learning can be divided into two categories according to different focuses: (1) to expanding capacity and (2) to improving security of steganography models.

2.1.1 Improving security

In 2016, Volkhonskiy et al. [22] firstly proposed a steganographic model called SGAN, which hides secret information into the cover image generated by DCGAN to improve the security of steganography. In 2017, Hayes and Danezis [23] constructed the encoder-decoder network called SteGAN to hide binary bitstream and extract it, and generated steganographic images via adversarial training. Tang et al. [13] proposed an automatic steganographic distortion learning framework—ASDL-GAN, which find the pixels in the carrier that are more secure steganography automatically. But the security of these models is still lower than the traditional adaptive steganography algorithms, and the steganographic capacity is limited to about 0.4 bpp or lower. In 2018, Yang et al. [15] exploit the Tanh-simulator instead of TES network based on ASDL-GAN in proposed UT-SCA-GAN, which solved the problem that the network cannot quickly converge during the training process. Meng et al. [24] proposed an improved traditional steganograhic algorithm via faster R-CNN, which could select specific steganographic locations for multiple steganographic algorithms; Zhu et al. [25] proposed a steganographic model called HiDDeN that is robust to noise, which introduces a noise layer to the encoder-decoder network to simulate a noise attack. Until 2019, Zhang et al. [16] introduced the critic network as a discriminator to provide feedback on the performance of encoder and reduce the distortion of steganograpic images. The capacity of this model was increased to 4.4 bpp. Tang et al. [14] proposed an adversarial embedding scheme based on CNN called ADV-EMB, which adjusts the partial modification costs of cover image according to the gradient propagated from the target CNN steganalyzer. But the capacity of this type of method is similar to the ordinary adaptive steganography algorithm. The security of this method became higher than traditional steganography algorithms.

2.1.2 Expanding capacity

In 2017, Baluja [17] proposed a steganographic model based on the encoder-decoder network, which could hide a color image into the same size of color image by the encoder network and revealed secret message by decoder network. Rehman [18] hid a grayscale image into another color image with the same size. But steganographic images generated by this kind of methods had obvious color distortion. In 2018, Wu et al. [26] proposed an end-to-end steganography model, which adds variant loss based on loss function built by Baluja to improve the visual quality of steganographic images. But the security of these

methods is poor. Zhang et al. [27] proposed an effective scheme to improve the ability of the adaptive steganography algorithm (WOW) to resist the detection of state-of-art steganalysis networks through adversarial examples. In 2019, Duan et al. [19] proposed an encoder network with structure similar to U-Net and generated steganographic images with high visual quality. Zhang et al. [20] proposed a steganographic model called ISGAN, which hides a grayscale image into the Y channel of the cover color image and improves the security of the model through adversarial training between the encoder-decoder network and steganalysis network.

2.2 Steganalysis based on deep learning

In 2015, Qian et al. [2] firstly proposed a steganalysis network called GNCNN, which utilized CNN (convolution neural network) to extract high-dimensional features and used the Gaussian function as the activation function. But the detection accuracy of these methods was lower than the rich model, which is currently the most widely used traditional steganalysis model. In 2016, Xu et al. [3] described a steganalysis model called Xu'Net that can be applied to the spatial and JPEG domains. Compared with GNCNN, ReLU function, ABS function, and batch normalization were added into the models to improve the detection accuracy. In 2017, Ye et al. [4] utilized 30 high-pass filters in SRM to initialize and introduce SCA (selection channel aware) to optimize the steganalysis model called Ye'Net. In 2019, Boroumand et al. [5] proposed a steganalysis model called SRNet constructed by residual layers with shortcut connections, which can encourage the reuse of features in the training process. The performance of this model is better than SCA-YeNet [4] and maxSRM [7]. However, SRNet is the hardest to train because of the vanishing gradient phenomenon.

3 The proposed method

In this chapter, we will introduce the proposed steganography method called HIGAN in detail from two aspects: the architecture of network and loss function. Compared with the existing steganography model used to hide the color images, our encoder network composed of downsampling layers, residual layers, and upsampling layers outputs the low color distortion and high visual quality steganographic images. At the same time, we introduce the steganalysis model as the discriminator to improve the security of model by the adversarial training with the encoder-decoder network.

3.1 The architecture of network

The architecture of the steganography model is shown in Fig. 1, which consists of three sub-networks: encoder, decoder, and discriminator. We use the encoder-decoder network, also known as the hiding-extraction network, as a generator to generate steganographic images and reveal secret images with low color distortion and high visual quality. The details of the above two sub-networks can be found in Sections 3.1.1 and 3.1.2, respectively. Then, the steganographic images obtained by the above sub-network are used as an input for discriminator network. And the security of the steganographic images is improved through adversarial training. The detailed description of the discriminator network can be seen in Section 3.1.3. In the training process, the distortion of steganographic image and reconstructed secret image and the ability to resist the detection of steganalysis models (discriminators) can be used as a measurement indicator to optimize

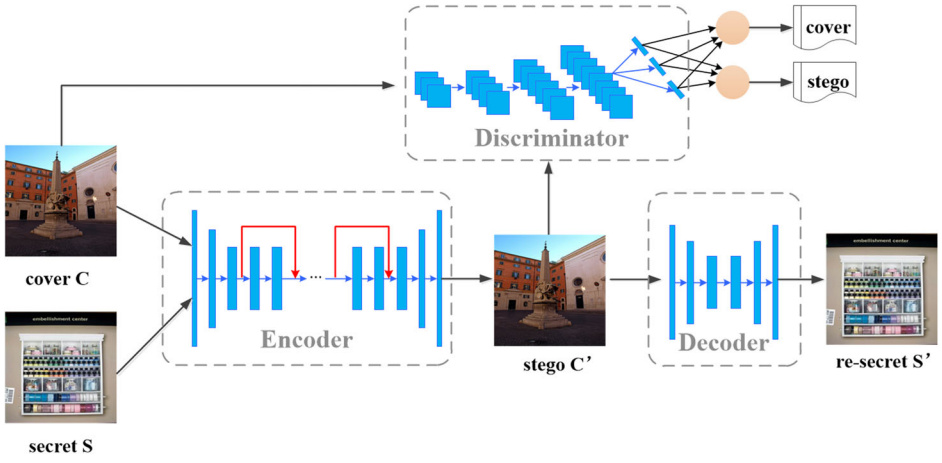

Fig. 1 The architecture of HIGAN. The whole steganography model is composed of sub-networks: encoder, decoder, and discriminator. Encoder could hide a secret color image into a cover color image with the same size. The steganographic images generated by encoder are used as the input of the decoder, and then, the secret images are extracted by decoder. Discriminator takes the cover images and steganographic images as input to distinguish whether the input images contain secret images

the weights of steganography model. The more detailed description of loss function can be seen in Section 3.2.

3.1.1 Encoder

Our encoder network consists of downsampling layers, residual layers, and upsampling layers. The structure of encoder network is shown in Fig. 2. In the resnet, it includes many skip connections to fuse the shallow features and deep features in the different convolution stages. The shallow feature includes a lot of low-level efficient information about the outline and color of image, which is very beneficial for the generation of steganographic images. The works of [19] constructed the encoder network similar to the structure of U-Net, and its results have proved that the skip connection is effective to reduce the distortion of steganographic images and improve the visual quality of steganographic images.

As we all know, the structure of GAN makes it extremely unstable during adversarial training process and it is often accompanied by phenomena such as gradient disappearance, which makes it difficult for the network to converge quickly. Therefore, it is necessary that we added nine residual blocks in the encoder network. Compared with the existing works used to hide color images [17, 19], our encoder network based on resnet can not only learn more rich feature information, but also reduce the disappearance of gradient during the progress of adversarial training. The specific details of the encoder network layers can also be seen in Table 1. Each residual block includes a skip connection and two convolutional layers with batch normalization operation and ReLU activation function, which can be seen in Table 2.

The shape of secret image S and cover image C is $D \times W \times H(3 \times 256 \times 256)$, where D is the channel of image, and $W \times H$ is the width and height of the image. The 6-dimensional tensor $6 \times 256 \times 256$ generated by concatenating the secret image S and the cover image C is used as the input of the encoder network. In the downsampling phase, it contains a 7×7 convolutional layer with stride 1 and two 3×3 convolutional layers

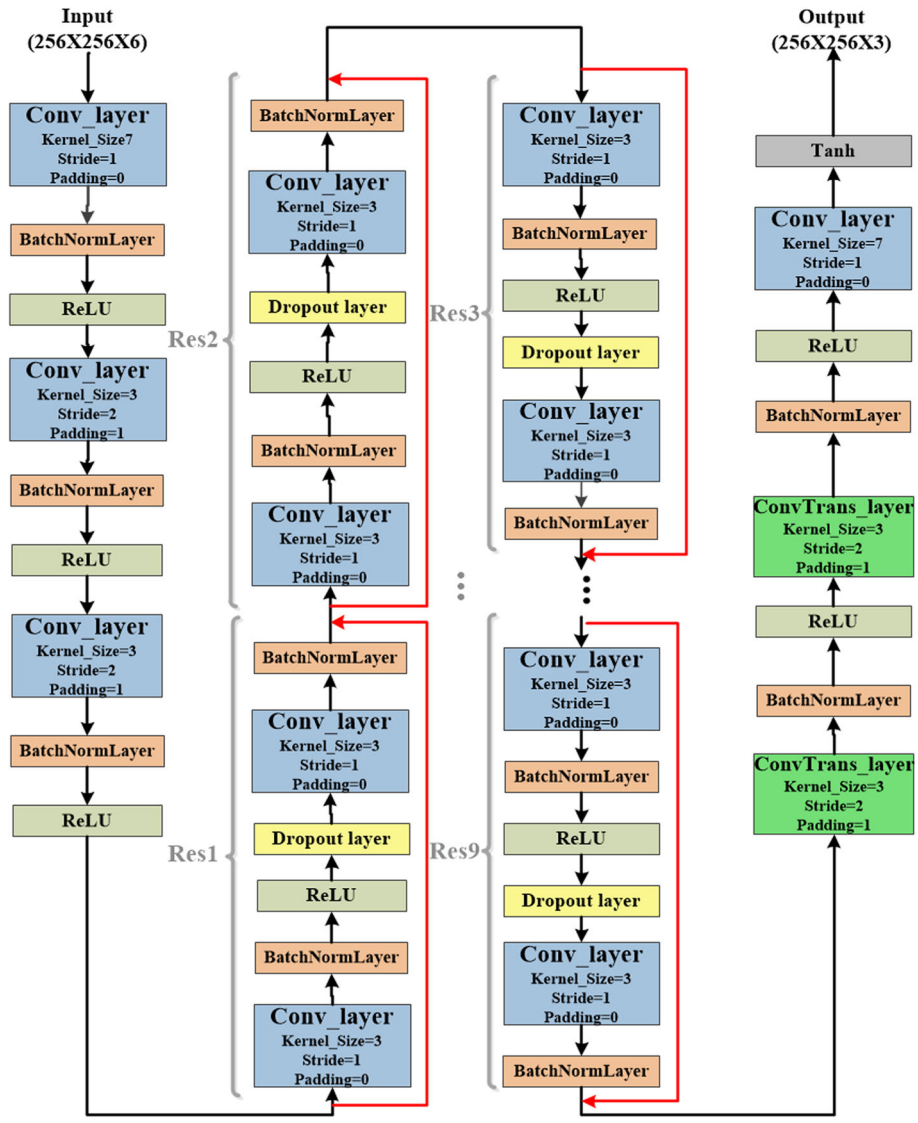

Fig. 2 The structure of encoder network The entire work is composed of three blocks: three downsampling layers, nine residual layers, and three upsampling layers. In the downsampling phase, the channels of image gradually increase from 6 to 256. In the residual layers, the channels and size of feature maps are maintained at 256 and 64. In the upsampling layers, the channels of feature maps gradually reduce from 256 to 3

with stride 2 and padding 1. Among them, each convolutional layer is followed by the batch normalization (BN) operation and the ReLU activation function. The third convolutional layer outputs 64×64 feature maps, which are the input of the continuous residual layers. In the middle residual layers, it contains nine continuous residual blocks to learn more rich low-dimensional and high-dimensional features. Each residual block includes two 3×3 convolutional layers with stride 1 and a skip connection. The dropout layer is added between the two convolutional layers. The input 64×64 feature map of each residual block is added to its output after two convolution operations, and the result of the addition operation is the final output of each residual block. In the upsampling phase, it contains two 3×3 deconvolutional layers with stride 2 and padding 1 and a 7×7 convolutional layer. Among them, the deconvolutional layers are followed by the BN operation

Table 1 Network architecture of proposed model

Encoder	Decoder	Discriminator (Xu'Net)
Input : 6 × 256 × 256	*Input* : 3 × 256 × 256	*Input* : 3 × 256 × 256
Conv(6 , 64 , 7)		
Conv(64, 128, 3)		
Conv(128, 256, 3)		
*Res_block*1(256, 256, 3)	*Conv*(3, 64, 3)	*kv_filter*(3, 1, 5)
*Res_block*2(256, 256, 3)	*Conv*(64, 128, 3)	*Conv*(1 , 8, 5)
*Res_block*3(256, 256, 3)	*Conv*(128, 256, 3)	*Conv*(8, 16 , 5)
*Res_block*4(256, 256, 3)	*Conv*(256, 128, 3)	*Conv*(16, 32 , 3)
*Res_block*5(256, 256, 3)	*Conv*(128, 64, 3)	*Conv*(32, 64, 3)
*Res_block*6(256, 256, 3)	*Conv*(64, 3 , 3)	*Conv*(64, 128, 3)
*Res_block*7(256, 256, 3)	*Sigmoid*(3, 3)	*Sigmoid*(128, 1)
*Res_block*8(256, 256, 3)		
*Res_block*9(256, 256, 3)		
DeConv(256 , 128, 3)		
DeConv(128 , 64 , 3)		
Conv(64 , 3, 7)		
Tanh(3, 3)		
Output : 3 × 256 × 256	*Output* : 3 × 256 × 256	*Output* : 1

Each resnet block includes two 3 × 3 convolutional layers with batch normalization operation and ReLU activation function, which outputs the result of adding the input feature map and the feature map after two convolutional layers

and ReLU activation function. Specifically, the last convolutional layer is followed by tanh activation function to output the 256×256 steganographic images C'. The implementation process of the encoder network can be expressed as follows:

$$Downsampling: \begin{cases} a = Conv_{6\to 64}(Cat(C,S)), \\ b = (Conv_{128\to 256}(Conv_{64\to 128}(a)), \end{cases} \tag{1}$$

$$Res-blocks: \begin{cases} a_1 = Conv_{256\to 256}(Conv_{256\to 256}(b)), \\ b_1 = b + a_1, \\ a_2 = Conv_{256\to 256}(Conv_{256\to 256}(b_1)), \\ b_2 = b_1 + a_2, \\ a_3 = Conv_{256\to 256}(Conv_{256\to 256}(b_2)), \\ b_3 = b_2 + a_3, \\ \quad \cdots \\ a_9 = Conv_{256\to 256}(Conv_{256\to 256}(b_8)), \\ b_9 = b_8 + a_9 \end{cases} \tag{2}$$

$$Upsampling: \begin{cases} a' = DeConv_{128\to 64}(\\ \quad DeConv_{256\to 128}(b_9) \\ b' = Conv_{64\to 3}(a') \end{cases} \tag{3}$$

Table 2 The PSNR and SSIM value of steganograhic images and revealed secret images

	Stego (SSIM)	Stego (PSNR)	Re-secret (SSIM)	Re-secret (PNSR)
Deep Steganography (Baluja)	0.92	28.41	0.92	28.06
Duan's model	0.95	36.71	0.96	36.97
HIGAN (our)	0.94	30.95	0.94	29.67

$$Stego\ image : E_n(C, S) = C' = Tanh(b') \tag{4}$$

3.1.2 Decoder

The decoder network composed of a 6-layer full convolutional network extracts the secret color images (S') from steganographic images (C'). The decoder network and the encoder network in the previous section constitute the generator of the proposed model HIGAN. The architecture of the decoder network has been proven in previous work to effectively reconstruct single-channel grayscale secret images and three-channel color secret images. All 3×3 convolutional layers with stride 1 and padding 1 are followed by batch normalization (BN) operation and ReLU activation function. But, sigmoid activation function was used after the last convolutional layers. Finally, the decoder network reveals the secret image S'. The implementation process of the decoder network can be expressed as below:

$$Revealed\ image : \begin{cases} a = Conv_{128\to256}(Conv_{64\to128} \\ \quad (Conv_{3\to64}(c'))), \\ b = Conv_{64\to3}(Conv_{128\to64} \\ \quad (Conv_{256\to128}(a))), \\ De(C') = S' = Sigmoid(b). \end{cases} \tag{5}$$

3.1.3 Discriminator

From the current works of Baluja [17] and Duan et al. [19], the steganography methods based on deep learning used to hide the color three-channel secret image just focused on improving the visual quality of steganographic images and revealed secret images. As we all know, security is one of the important indicators to measure the quality of steganography models. However, this type of steganography model has poor security due to too much information embedded in the carrier image. In view of the above problem, the literature [20], proposed an improved steganography model that hid the secret gray-scale image in the Y channel of the color carrier image. Even if the embedding capacity is high, the security still can be improved by the adversarial training with steganalysis model. The steganalysis model can be used as a binary discriminator to determine whether the cover images contain secret information. Therefore, the advanced steganalysis model Xu'Net, which is used as a discriminator in our model. Its detection accuracy for adaptive steganography algorithms was higher than SRM. The architecture details of the discriminator can be seen in Table 1. In order to make it suitable for detecting color images, we transpose the KV kernel of the first layer of Xu'Net to make it a three-channel 5×5 matrix. The structure of the remaining five convolutional layer remains unchanged. In the last fully connected layers, we used the sigmoid activation function to output the detection probability, and its value range is (0, 1). The implementation process of the discriminator network can be expressed as below:

$$\begin{cases} a = Conv_{16\to32}(Conv_{3\to16} \\ \quad (Conv_{3\to3}(C\ or\ C'))), \\ b = AvgPool_{128}(Conv_{64\to128} \\ \quad (Conv_{32\to64}((a))), \\ Dis(C\ or\ C') = Sigmoid(b). \end{cases} \tag{6}$$

3.2 Loss function

We optimize the weights of the encoder-decoder network through adversarial training. Firstly, we use MSE (mean square error) loss to measure the similarity between the original images (such as cover images C, secret images S) and reconstructed images (such as steganographic images C', revealed images S'). This loss has been used in the training of the proposed steganographic models based on encoder-decoder network, such as Deep Steganography [17] and ISGAN [20]. Moreover, we use the binary cross-entropy loss as the possibility that the steganographic image can be detected as the carrier image category. The loss functions are as follows:

$$L_en(C, C') = \mathbb{E}_{X\sim\mathbb{P}_c}\frac{1}{3\times W\times H}\|C - En(C, S)\|_2^2 \tag{7}$$

$$L_s(S, S') = \mathbb{E}_{X\sim\mathbb{P}_c}\frac{1}{3\times W\times H}\|S - De(C')\|_2^2 \tag{8}$$

$$L_r(C', y_t) = \mathbb{E}_{X\sim\mathbb{P}_c} BinaryCrossEntropy(C', y_f) \tag{9}$$

where the $En(\cdot)$, $De(\cdot)$, and y_t represent the encoder network, the decoder network, and the label of cover images. The total loss function can be expressed as:

$$minmize L_{total} = L_s(C, C') + \beta L_s(S, S') + \gamma L_r(C', y_t) \tag{10}$$

where β, γ are the weight parameters that can be set artificially, which represents the weight of each different loss in the total loss function. We set it to 0.75 and 1 in the experiment, respectively. To train the discriminator network, we consider the encoder-decoder network as generator. Generator and discriminator play a two-player minmax game [21].

$$\begin{aligned} L_{GAN} =& E_{C\sim P(c)}[\,logD(C)] + E_{C\sim P(C),S\sim P(S)} \\ & [\,log(1 - D(E_n(C, S)))] \end{aligned} \tag{11}$$

4 Experiments

In this section, we will introduce the implementation process and details of experiments which include the details of datasets, the parameter settings, and the evaluation metrics.

4.1 Dataset

We use the validation sets of ImageNet2012 [28] as the training, validation, and test sets of our experiments, which contain 50,000 images from 1000 categories. We randomly choose 40,000 images to form 20,000 pairs of cover-secret images as the training set. From the remaining 10,000 pictures, we randomly selected 5000 pictures to form the 2500 pairs of cover-secret images of the validation set, and the other 5000 pictures to form 2500 pairs of cover-secret images of the test set. The GPU is TITAN RTX, the experimental environment is Pytorch, and the application is Python 3.6.

4.2 Implementation processes and details

In the traditional training process of generative adversarial networks, the generator and the discriminator are alternately trained. When one of the networks is trained, the weight of the other is fixed and will not be updated. The training sequence is to train the discriminator first and then the generator. In order to speed up the convergence of the model, the weights of the discriminator are usually updated every 5 times, and then, the weights of

the generator are updated once. Therefore, we use the encoder-decoder network as a generator and the steganalysis model as a discriminator. Then, we perform alternate training between the generator and discriminator. Due to the fact that too much information is embedded in our carrier image, the euclidean distance between the steganographic image and the carrier image is long. So the steganalysis model Xu'Net is easy to converge. To balance the performance of encoder-decoder network and discriminator, we update the weights of generator every 4 times and then update the weights of discriminator once. The total number of training epochs is set as 200.

In the training process of encoder and decoder network, the initial learning rate (lr) is set as 0.001, and the Adam optimization method is used to update the parameters of the encoder and decoder network automatically. And we set an initial learning rate of discriminator as 0.001/2 and use the SGD optimization method to update the parameters of discriminator. Moreover, we reduce the learning rate of the three sub-networks to $0.9 * lr$ each 3 epochs.

4.3 Evaluation metrics

In general, there are three criteria for measuring steganography algorithms; it includes the capacity, security, accuracy of extraction. Firstly, capacity represents the number of secret information to be embedded in the carrier image and it can be reconstructed after embedding. Secondly, security not only represents the ability of resisting the detection of steganalysis model, but also means the reality and low distortion of steganographic images and the invisibility of secret images. Even if the steganographic image generated by the steganography model can escape the detection of the steganographic analysis model, it is still not feasible that it can be found abnormal and unnatural by the naked eye. Finally, extraction accuracy is very important for the effectiveness of steganographic models. No matter how much secret information is embedded in the carrier image, if it cannot be extracted completely and accurately, then the whole steganography method is meaningless. Currently, all steganography methods based on deep learning have these problems in the extraction accuracy, that is, it cannot reconstruct the secret information perfectly. However, the steganography method used to hide images has a unique advantage that it does not require decoder network to recover the secret image perfectly. Even if the secret image is not perfectly reconstructed, the receiver can still understand the whole semantic information contained in the secret image.

4.3.1 Peak signal to noise ratio (PSNR)

This metric evaluates the visual quality of images by calculating the error between corresponding pixels and has shown to be correlated with mean opinion scores produced by human experts. It is defined as a function of MSE. The larger the values, the smaller the distortion of images after steganography. The PSNR [29] can be computed by using:

$$MSE = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} \left(X_{i,j} - Y_{i,j}\right)^2 \tag{12}$$

$$PSNR = 10 \cdot \log_{10} \left(\frac{(2^n - 1)^2}{MSE} \right) \tag{13}$$

4.3.2 Structural similarity index (SSIM)

This metric measures the quality of steganographic images in three ways: brightness, contrast, and structure. The larger the value of SSIM, the higher the similarity of images before and after steganography. The SSIM [29] can be computed by using:

$$SSIM(X, Y) = l(X, Y) \cdot c(X, Y) \cdot s(X, Y) \tag{14}$$

$$l(X, Y) = \frac{2\mu_x\mu_y + C_1}{(\mu_x)^2} + (\mu_y)^2 + C_1 \tag{15}$$

$$c(X, Y) = \frac{2\sigma_x\sigma_y + C_2}{(\sigma_x)^2} + (\sigma_y)^2 + C_2 \tag{16}$$

$$s(X, Y) = \frac{2\sigma_{xy} + C_2}{\sigma_x\sigma_y} \tag{17}$$

where $l(\cdot)$, $c(\cdot)$, and $s(\cdot)$ represent luminance, contrast, and structure, respectively. μ_X and μ_Y are the pixel average of image X and image Y, respectively; σ_X and σ_Y represent the standard deviation of image X and image Y; $\sigma_X\sigma_Y$ denotes the covariance of image X and image Y.

4.3.3 Pixel error

This metric quantifies the difference between the corresponding pixel values of the two images. Therefore, it could reflect the reconstruction error of cover image and secret images [17]. The larger the values, the larger the reconstruction error. The pixel error can be computed as:

$$Pixel\_error(X, Y) = \sum_{i=1}^{W}\sum_{j=1}^{H}\left(X_{i,j} - Y_{i,j}\right)^2 \tag{18}$$

5 Results and discussion

We carry out the comparison experiments with the existing works from three respects: (1) the visual quality of steganographic images and revealed secret images, (2) the security of steganographic methods, and (3) the accuracy of reconstruction. To control the variables, we use the ImageNet2012 dataset to retrain all relevant steganography models based on deep learning used to hide color images. Before the beginning of all comparison experiments, 1000 pairs of cover-stego images and 1000 pairs of secret images and revealed secret images were generated by each retrained comparison models, respectively, as the test set, which is used to measure the performance of these models.

5.1 The quality of steganographic and revealed images

The steganographic images and revealed secret images generated by our model are shown in Fig. 3. And the results of existing works [17, 19] are shown in Figs. 5 and 6, respectively. From the displayed images in the Fig. 3, even if a color image is embedded in the carrier image with the same size, the steganographic images generated by our model still do not have the problem of color deviation and peak noise point. Compared with the existing steganography model [17] in the second row of Fig. 6, the visual quality of the steganographic image generated by our model is higher. But compared to another steganography model [19] in the second row of Fig. 5 (its results are shown in Fig. 5), the outlines and edges of the steganographic image generated by our encoder are more

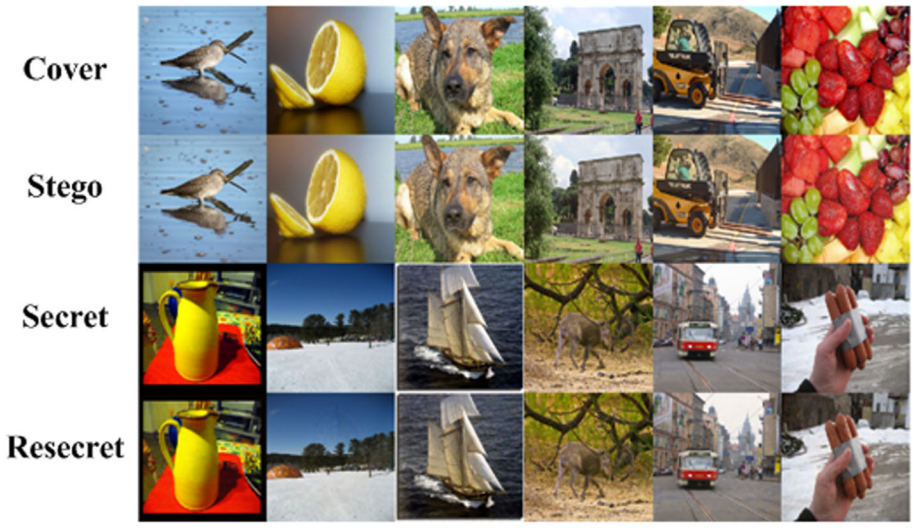

Fig. 3 The performance of our model. The first row in the figure is the cover images, the second row is the corresponding steganographic images , the third row is the secret images, and the last row is the reconstructed secret images

blurred than the steganographic image generated by its encoder. We analyze that this is because the structure of encoder is similar to U-Net, which includes many of the cascade operation. It enables their model to repeatedly learn low-dimensional features from the different shallow convolutional layers, such as outline and edge of image.

Table 2 shows the visual quality of steganographic images and reconstructed secret images generated by our model and other comparison model. The higher the visual quality means that the carrier image with secret information is more realistic and the difference with the original carrier image is smaller. To control variables, we retrain the Deep Steganography on the validation set of ImageNet2012 to make it suitable for generating the 256×256 steganographic images and reconstructing the 256×256 secret images. From the results shown in Table 2, we can see that the SSIM values of steganographic images and reconstructed secret images have improved by 2% compared with the Deep Steganography [17]. But the SSIM value of steganographic images is still 1% lower than Duan's model, and the SSIM value of reconstructed secret images is 2% lower than Duan's model. Therefore, we conclude that the fusion of feature maps of different layers as the input of the deep convolutional layer, and the repeated learning of the low-dimensional features of the shallow convolutional layer output in the deep convolutional layer are two methods to improving the visual quality of steganographic images effectively. However, too many jump connections will cause the features of the secret image to be fused with the features of the carrier image too closely, which will increase the difficulty of extracting the decoder network.

5.2 The security of our method

We measure the security of the model from two aspects: the invisibility of the secret image and the ability of the steganographic image to resist the detection of steganalysis model. We show that the residual image is obtained by subtracting the corresponding pixel values of the steganographic image and the carrier image with $5\times$, $10\times$, and $20\times$ enhancement

in Figs. 4, 5, and 6. In the residual images with 5× enhancement, almost nothing about secret images was visible in all comparison models in the fourth row of Figs. 4, 5, and 6. Then, when we reexamine the residual image with 20× enhancement, the features of secret image are revealed on the residual image in the sixth row of Figs. 4, 5 and 6. But, still nothing about secret image is visible on our residual images in the sixth row of Figs. 4 and 5, which means the strong invisibility of secret images in the steganographic images.

Moreover, we changed the structure of first layer in the two steganalysis models Xu'Net [3] and SRNet [5] to make them suitable for detecting the three-channel color images. Then, we use 4000 pairs of carrier and steganographic images generated by the Deep Steganography [17] as training set and validation set. Finally, Xu'Net trained for 100 epochs and SRNet trained for 20 epochs are used as our steganalytic tools. The difference in the number of training epochs is due to the different convergence speeds of the two models. The detection of accuracy is shown in Table 3. The detection accuracy of Xu'Net for the 1000 steganographic images generated by our model is only 39.7%, and the detection accuracy of SRNet for the 1000 steganographic images generated by our model is only 33.4%. Compared with the works of [17] and [19], the detection accuracy is reduced by average 50%, and the ability of resisting the detection of steganalysis model is significantly improved. Therefore, it also proves again that steganogarphy model can effectively improve its security through adversarial training.

5.3 The accuracy of reconstruction

Due to the operations such as batch normalization and pooling in the training process of network, all steganography models based on encoder-decoder network cannot extract the

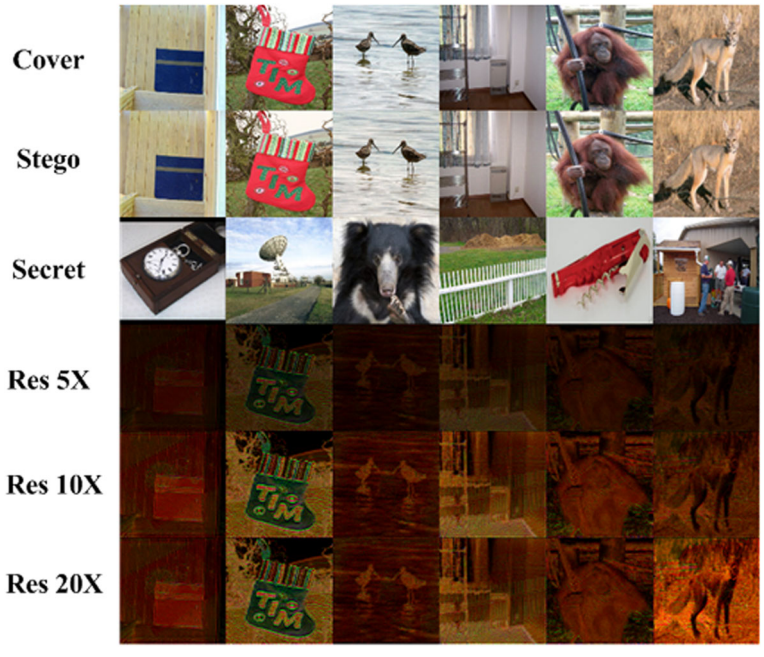

Fig. 4 The residual steganographic images generated by HIGAN. The first three rows are the cover images, the corresponding steganographic images generated by our model, and the secret images, respectively; the last three rows are the residual images obtained by the subtraction of the steganographic images and corresponding carrier images with 5×, 10×, and 20× enhancement

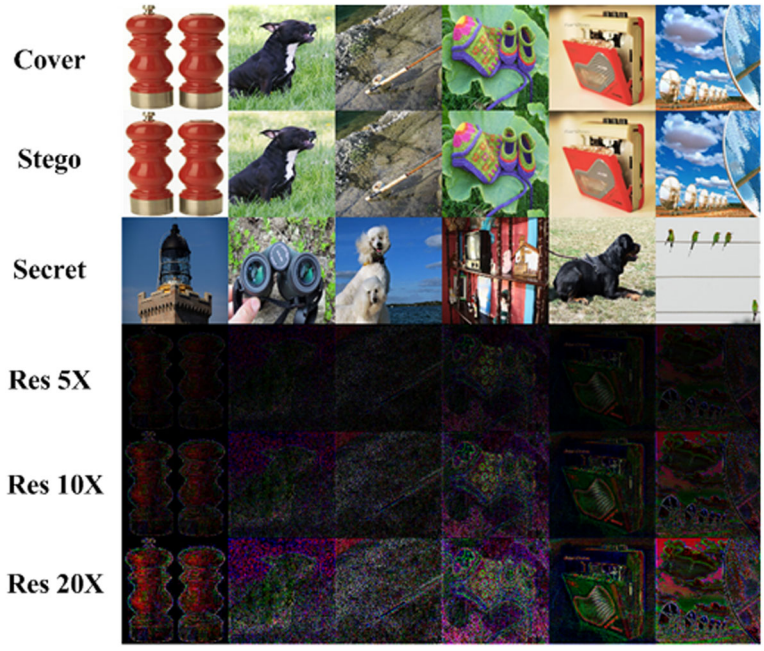

Fig. 5 The residual steganographic images generated by Duan's model. The first three rows are the cover images, the corresponding steganographic images generated by Duan's model, and the secret images; the last three rows are the residual images obtained by subtracting the pixel values of the steganographic images and corresponding cover images with 5×, 10×, and 20× enhancement

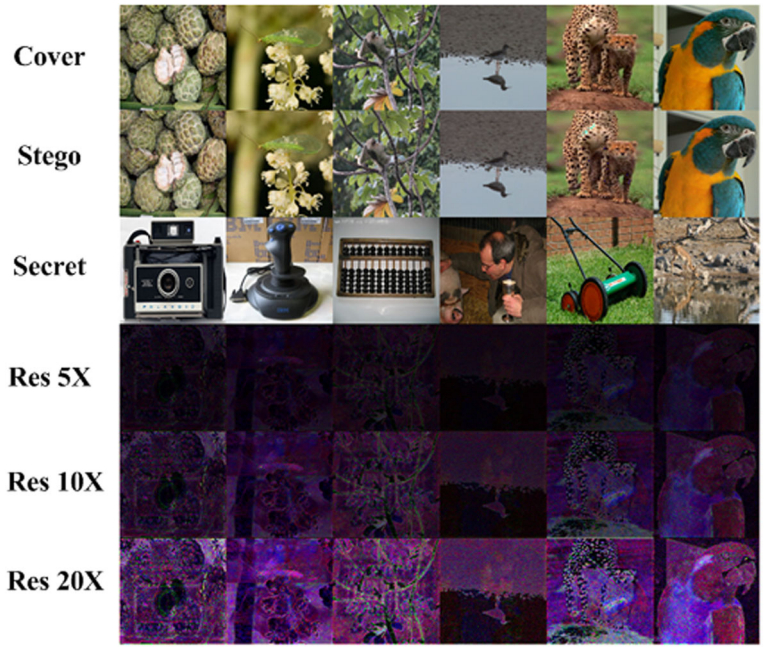

Fig. 6 The residual steganographic images generated by Deep Steganography The first three rows in the figure are the cover images, the corresponding steganographic images generated by Deep Steganography, and the secret images; the last three rows are the residual images obtained by the subtraction of the steganographic images and corresponding cover images with 5× , 10×, and 20× enhancement, respectively

Table 3 The detection accuracy of steganalysis model

	Deep Steganography [17]	Duan's model [19]	HIGAN (our)
Xu'Net [3]	39.7%	71.1%	98.9%
SRNet [5]	33.4%	82.4%	99.5%

secret information perfectly. For the steganography methods used to hide the binary bitstream, a little error of extraction may cause the final reconstructed secret information to be very different from the original secret information. However, the above situation hardly happens for the steganography methods used to hide the secret images. The deviation or loss of some pixel values will not affect the receiver's understanding of the semantic information contained in the secret images. It should be noted that some redundant information in the natural image that is regarded as secret information is ignored, because the amount of redundant information in different natural images is different and does not affect the extraction. Therefore, the capacity of our model is almost closely to 24 bpp (it is equivalent to that we hide three pixels (24 bits) in a pixel, because the embedded secret information is the three-channel color image)

But we still use quantified results to measure the accuracy of reconstruction. The pixel error calculated by sse represents the accumulation of the difference in the corresponding pixel values between the reconstructed secret image and the original secret image. The larger the values of pixel error, the worser the accuracy of extraction. The total reconstruction error of the secret images and cover images for three channels is 7.95 and 5.87, respectively, in the test set, which includes 2500 pairs of three-channel cover-stego images and 2500 pairs of three-channel reconstructed secret images and revealed secret images generated by our model. In the existing works of [17], the pixel error of the secret image and cover image was 3 and 3.2, respectively, for each channel. Moreover, the histogram of errors for the cover and reconstructed cover (stego) is shown on the right side of Fig. 7. And the histogram of errors for the secret and reconstructed secret is shown on the left side of Fig. 7. As can be seen, there are few large pixel errors.

6 Conclusion

In this paper, we propose a steganographic model called HIGAN, which is composed of three sub-networks: the encoder network, the decoder network, and the discriminator network (steganalysis model). It can encode the secret color image into another color

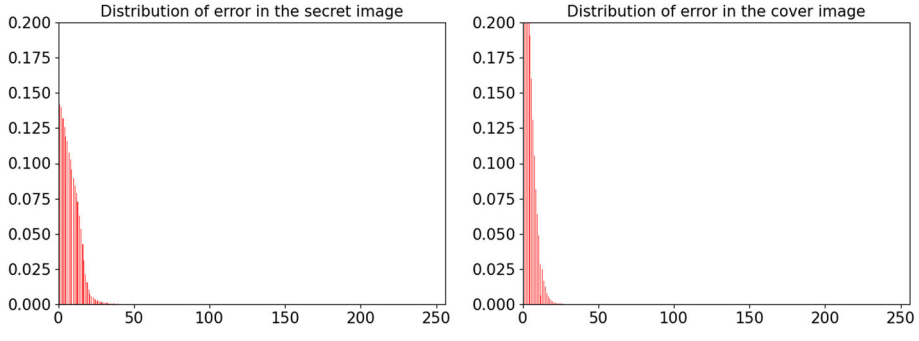

Fig. 7 The distribution of pixel errors for 3-channel cover images and secret images, respectively. The horizontal axis of the histogram represents the value of the pixel error, and the vertical axis represents the frequency of the corresponding pixel error

image with the same size to generate the steganographic images with high visual quality by encoder network, which is composed of the nine residual blocks. And secret images are extracted by decoder network. Besides, the security of the steganography model is improved by adversarial training between encoder-decoder network and discriminator network. The experimental results show that steganographic images generated by our model have less color distortion and better steganography security.

Abbreviations

SRM: Spatial rich model; GAN: Generative adversarial network; SPP: Spatial pyramid pooling; S-UNIWARD: Spatial universal wavelet relative distortion; HILL: High-pass, low-pass, and low pass; CNN: Convolutional neural network; WOW: Wavelet obtained weights; BN: Batch normalization; SCA: Selection channel aware; MSE: Mean square error; PSNR: Peak signal to noise ratio; SSIM: Structural similarity index

Acknowledgements

Thanks to my tutor and classmates for their help in collecting experimental datasets and writing the article.

Authors' contributions

The analysis of the experimental results and the refining of the article are completed by ZF. The first draft of the paper and experiments are completed by FW. The authors read and approved the final manuscript.

Authors' information

Zhangjie Fu is currently a Professor of Computer Science and the Director of Bigdata Security Lab at Nanjing University of Information Science and Technology, China. His research interests include data security, digital forensics, and network and information security. Fan Wang is currently a master in the Nanjing University of Information Science and Technology, China. Her research interests include network and information security, information hiding, and deep learning. Xu Cheng is currently an associate professor in Nanjing University of Information Science and Technology. His research interests include computer vision, pattern recognition, and image processing.

Funding

This work is supported by the National Natural Science Foundation of China under grant U1836110 and the National Natural Science Foundation of China under grant 61801058.

Availability of data and materials

Data and implementation codes for all experiments are based on python. The datasets used during the current study are available from the corresponding author on reasonable request.

Competing interests

The authors declare that they have no competing interests.

Author details

[1] Department of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China. [2] Pengcheng Laboratory, Shenzhen 518000, China.

Received: 5 May 2020 Accepted: 23 September 2020
Published online: 27 October 2020

References

1. F. A. Petitcolas, R. J. Anderson, M. G. Kuhn, Information hiding-a survey. Proc. IEEE. **87**(7), 1062–1078 (1999)
2. Y. Qian, J. Dong, W. Wang, T. Tan, in *Proceedings of Media Watermarking, Security, and Forensics 2015: 9-11 February 2015; San Francisco*. ed. by M. A. Adnan, D. M. Nasir, and D. H. Chad, Deep learning for steganalysis via convolutional neural networks (International Society for Optics and Photonics, 2015), pp. 171–180
3. G. Xu, H.-Z. Wu, Y.-Q. Shi, Structural design of convolutional neural networks for steganalysis. IEEE Signal Proc. Lett. **23**(5), 708–712 (2016)
4. J. Ye, J. Ni, Y. Yi, Deep learning hierarchical representations for image steganalysis. IEEE Trans. Inf. Forensics Secur. **12**(11), 2545–2557 (2017)
5. M. Boroumand, M. Chen, J. Fridrich, Deep residual network for steganalysis of digital images. IEEE Trans. Inf. Forensics Secur. **14**(5), 1181–1193 (2018)
6. J. Fridrich, J. Kodovsky, Rich models for steganalysis of digital images. IEEE Trans. Inf. Forensics Secur. **7**(3), 868–882 (2012)
7. T. Denemark, V. Sedighi, V. Holub, R. Cogranne, J. Fridrich, in *Proceedings of the 2014 IEEE International Workshop on Information Forensics and Security: 3-5 December 2014; Atlanta*. ed. by Y. Sun, V. H. Zhao, Selection-channel-aware rich model for steganalysis of digital images (IEEE, 2014), pp. 48–53
8. Y. Yang, Y. Chen, Y. Chen, W. Bi, A novel universal steganalysis algorithm based on the iqm and the srm. Comput. Mater. Continua. **56**(2), 261–272 (2018)
9. X. Duan, H. Song, C. Qin, M. K. Khan, Coverless steganography for digital images based on a generative model. Comput. Mater. and Continua. **55**(3), 483–493 (2018)
10. V. Holub, J. Fridrich, T. Denemark, Universal distortion function for steganography in an arbitrary domain. EURASIP J. Inf. Secur. **2014**(1), 1 (2014)

11. V. Holub, J. Fridrich, in *Proceedings of the 2012 IEEE International Workshop on Information Forensics and Security: 2-5 December 2012; Tenerife*. ed. by P. Moulin, F. Pérez-González, Designing steganographic distortion using directional filters (IEEE, 2012), pp. 234–239
12. B. Li, M. Wang, J. Huang, X. Li, in *Proceedings of the 2014 IEEE International conference on Image Processing: 27-30 October 2014; Paris*. ed. by B. Pesquet-Popescu, J. Fowler, A new cost function for spatial image steganography (IEEE, 2014), pp. 4206–4210
13. W. Tang, S. Tan, B. Li, J. Huang, Automatic steganographic distortion learning using a generative adversarial network. IEEE Signal Proc. Lett. **24**(10), 1547–1551 (2017)
14. W. Tang, B. Li, S. Tan, M. Barni, J. Huang, CNN-based adversarial embedding for image steganography. IEEE Trans. Inf. Forensics Secur. **14**(8), 2074–2087 (2019)
15. J. Yang, K. Liu, X. Kang, E. K. Wong, Y.-Q. Shi, Spatial image steganography based on generative adversarial network. arXiv preprint arXiv:1804.07939 (2018). http://arxiv.org/abs/1804.07939
16. K. A. Zhang, A. Cuesta-Infante, L. Xu, K. Veeramachaneni, SteganoGAN: high capacity image steganography with GANs. arXiv preprint arXiv:1901.03892 (2019). http://arxiv.org/abs/1901.03892
17. S. Baluja, in *Proceedings of Advances in Neural Information Processing Systems 30(NIPS 2017): 3-9 December 2017; Long Beach*. ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Hiding images in plain sight: deep steganography (Curran Associates, Inc, 2017), pp. 2069–2079
18. A. Rehman, R. Rahim, S. Nadeem, S. Hussain, S. Roth, in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops: 8-14 September 2018; Munich*. ed. by Leal-Taixé L., End-to-end trained cnn encoder-decoder networks for image steganography (Springer International Publishing, 2019), pp. 723–729
19. X. Duan, K. Jia, B. Li, D. Guo, E. Zhang, C. Qin, Reversible image steganography scheme based on a U-Net structure. IEEE Access. **7**, 9314–9323 (2019)
20. R. Zhang, S. Dong, J. Liu, Invisible steganography via generative adversarial networks. Multimed. Tools Appl. **78**(7), 8559–8575 (2019)
21. I. Goodfellow, J. Pouget-Abadie, M. Mehdi, B. Xu, D. Warde-Farely, S. Ozair, A. Courvile, Y. Bengio, in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops: 8-14 September 2018; Munich*. ed. by Z. Ghahramani, M. Welling, C. Cotes, N. D. Lawrence, and K. Q. Weinberger, Generative adversarial nets (Curran Associates, Inc., 2019), pp. 723–729
22. D. Volkhonskiy, B. Borisenko, E. Burnaev, Generative adversarial networks for image steganography (2016). https://openreview.net/references/pdf?id=HJODCvqex
23. J. Hayes, G. Danezis, in *Proceedings of Advances in Neural Information Processing Systems 30(NIPS 2017): 3-9 December 2017; Long Beach*. ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Generating steganographic images via adversarial training (Curran Associates, Inc., 2017), pp. 1954–1963
24. R. Meng, S. G. Rice, J. Wang, X. Sun, A fusion steganographic algorithm based on faster R-CNN. Comput. Mater. Continua. **55**(1), 1–16 (2018)
25. J. Zhu, R. Kaplan, J. Johnson, F. Li, in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops: 8-14 September 2018; Munich*. ed. by V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Hidden: hiding data with deep networks (Springer International Publishing, 2018), pp. 657–672
26. P. Wu, Y. Yang, X. Li, Stegnet: mega image steganography capacity with deep convolutional network. Future Internet. **10**(6), 54 (2018)
27. Y. Zhang, W. Zhang, K. Chen, J. Liu, Y. Liu, N. Yu, in *Proceedings of the 6th ACM Workshop on Information Hiding and Multimedia Security: 20-22 June 2018; Innsbruck*. ed. by R. Böhme, C. Pasquini, Adversarial examples against deep neural network based steganalysis (Association for Computing Machinery, 2018), pp. 67–72
28. J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, F. Li, in *Proceedings of the 2009 IEEE Conference Vision and Pattern Recognition: 20-25 June 2009; Miami*. ed. by D. Huttenlcoher, G. Medioni, and J. Rehg, Imagenet: a large-scale hierarchical image database (IEEE, 2009), pp. 248–255
29. Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)

Publisher's Note