

Received February 23, 2020, accepted March 11, 2020, date of publication March 25, 2020, date of current version April 9, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2983175

Recent Advances of Image Steganography With Generative Adversarial Networks

JIA LIU¹, YAN KE¹, ZHUO ZHANG¹, YU LEI, JUN LI, MINQING ZHANG, AND XIAOYUAN YANG

Laboratory of Network and Information Security, Engineering University of People Armed Police Force, Xi'an 710086, China

Corresponding author: Jia Liu (liujia1022@gmail.com)

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFB0802000, in part by the National Natural Science Foundation of China under Grant U1636114, Grant 61772550, Grant 61572521, Grant 61379152, and Grant 61403417, and in part by the National Cryptography Development Fund of China under Grant MMJJ20170112.

ABSTRACT In the past few years, the Generative Adversarial Network (GAN), which proposed in 2014, has achieved great success. There have been increasing research achievements based on GAN in the field of computer vision and natural language processing. Image steganography is an information security technique aiming at hiding secret messages in common digital images for covert communication. Recently, research on image steganography has demonstrated great potential by introducing GAN and other neural network techniques. In this paper, we review the art of steganography with GANs according to the different strategies in data hiding, which are cover modification, cover selection, and cover synthesis. We discuss the characteristics of the three strategies of GAN-based steganography and analyze their evaluation metrics. Finally, some existing problems of image steganography with GAN are summarized and discussed. Potential future research topics are also forecasted.

INDEX TERMS Image steganography, generative adversarial nets, cover synthesis, generative model.

I. INTRODUCTION

Steganography is the art of hiding a secret message behind the normal message. The term steganography is also known as secret writing [1]. As for cryptography, the distinct visible encrypted information, no matter how unbreakable, will attract more attention of attackers. Steganography offers a feasible alternative to encryption in oppressive regimes where using cryptography might attract unwanted attention [2]. Classical steganography refers to the means of secret communication used by people in ancient times, mainly including invisible ink, Cardan grille, Tibetan poetry, and so on. Modern steganography refers to the use of electronic communication and digital technology to hide the message into digital media. Every modern steganographic system consists of two essential components: the embedding and extraction algorithms. The embedding algorithm accepts three inputs: the secret message, the secret key, and the cover object, which will be used to convey the message. The output of the embedding algorithm is called the stego object. The stego object is also presented as an input to the extraction algorithm for producing the secret message. All concepts and methods

presented in this paper are illustrated on the example of digital images.

In modern steganography [1], three basic principles for constructing steganographic methods are introduced. a) Steganography by cover modification (CMO). Steganographer starts with a cover image and makes modifications to it in order to embed secret data. However, the modification will inevitably introduce some embedding changes into the cover image. b) Steganography by cover selection (CSE), which is similar to image retrieval. Steganographer selects a natural, unmodified, normal image in an extensive database that can extract messages as a stego image. This method has a very low payload so that it cannot be applied to practical applications. c) Steganography by cover synthesis (CSY). Steganographer creates a stego image containing secret messages. However, about ten years ago, constructing a realistic digital image is more a theoretical construct rather than a practical steganographic technique.

Fortunately, a generative model, generative adversarial network, was proposed in 2014 [3]. The generator in the GAN model is capable of synthesizing realistic images. There are two main directions for the study of GAN. One trend is to optimize the model of [4]–[7] from different aspects, such as information theory [8] and an energy-based model [9].

The associate editor coordinating the review of this manuscript and approving it for publication was Fan-Hsun Tseng¹.

The other research direction is to try to apply GAN to more research fields, such as computer vision (CV) [4] and natural language processing (NLP) [10]. [11]–[13] reviews recent GAN models and applications. However, But these review papers do not focus on a specific application.

Recently, there are many pieces of research using GAN to design steganography schemes [14]–[19]. In this paper, we focus on GAN’s research progress in a particular field of image steganography. The primary purpose of this paper is to try to discuss the role of GAN in these image steganography methods and point out the problems faced by GAN-based image steganography. Current steganography methods using GAN have covered the three traditional strategies, i.e., modification methods, selection methods, and synthesis methods. Besides these GAN-based methods, there are some other ways to design steganography schemes using deep neural networks [20] or adversarial samples [21], which will be mentioned briefly. To the best of our knowledge, this is the first paper that attempts to review the applications of GAN in image steganography.

In this paper, we start with the basic model, conventional techniques, and security issues for steganography. After introducing the basic concept of GANs, we give a discussion of the improvements and applications of GAN. Then, we focus on the role of GAN in steganography. GAN-based steganography is also divided into three categories. They are cover modification (GAN-CMO), cover selection (GAN-CSE), and cover synthesis (GAN-CSY). In GAN-CMO, GAN is used to generate the cover image, learn modification strategy, or fool a steganalysis classifier. In GAN-CSE, the generator trained by GAN is used to build the mapping between the message and the cover image. In GAN-CSY, GAN is used to generate a stego image. Furthermore, according to the different dependence on the cover image when creating a stego image, the GAN-CSY methods can be divided into supervised methods, semi-supervised methods, and unsupervised methods. This division enables us to have a better understanding of GAN’s role in steganography by cover synthesis. Besides, we also analyze the characteristics of the GAN-CSY methods and summarize some general rules in designing steganography by cover synthesis. We also discuss some evaluation criteria of the GAN-based steganography, including security, steganographic capacity.

The rest of this paper is organized as follows, the classical steganography model, strategies, and security criteria are introduced in Section II. A brief review of the implementation of traditional steganography schemes is given, focusing on the characteristics and performance of these methods. In Section III, we briefly review the basic concept and applications of GAN. Then In Section IV, we discuss several methods in cover modification with GAN. In Section V, a special cover selection method using GAN is discussed. In Section VI, a detailed discussion is provided on the role of GAN in steganography by cover synthesis. In Section VII, we provide some evaluation metrics for image steganography

by GAN. A short conclusion and perspective with some possible research directions are given in Section VIII.

II. STEGANOGRAPHY PRELIMINARIES

A. STEGANOGRAPHY MODEL

The classical steganographic model is the prisoner’s problem [22] with three participants, as illustrated in Fig.1. Both Alice and Bob are held in separate cells. They are allowed to communicate with each other, but all their communications are monitored by the warden Wendy. In modern steganography, every channel between Alice and Bob contains five elements: cover source c , message embedding/extraction algorithm Emb/Ext, a secret key k for embedding/extraction, secret message m , and communication channel.

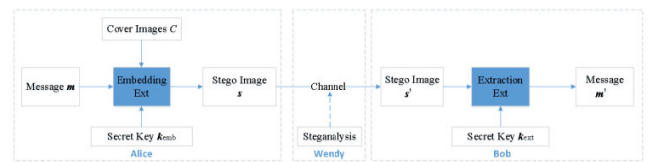


FIGURE 1. The prisoner’s problem model for steganography.

Using a data embedding method $Emb(\cdot)$, based on a specific cover image c or a set of cover images C , Alice needs to design a scheme to construct a stego image s with an embedding key k_{emb} . The stego images s can be expressed as:

$$s = Emb(c|C, m, k_{emb}) \tag{1}$$

For Bob, the stego image he receives can be expressed as s' . He can recover a secret message m' using an extraction key k_{ext} , and message extraction operation $Ext(\cdot)$.

$$m' = Ext(s', k_{ext}) \tag{2}$$

The message extraction key and the embedded key can be different from public key steganography [23]. In this paper, we only focus on the symmetric steganographic algorithm, where $k_{emb} = k_{ext}$ is assumed. When $s' = s$ is guaranteed, the steganographic channel is lossless. The above Eq. (1) and Eq. (2) only describe the process of message embedding and extraction. For the steganographic task, the core requirement is that the stego image s must be indistinguishable from the cover image c or the cover set C to realize the mission of information hiding. Here we define an abstract distance metric $D_{distinguishable}$ to represent indistinguishability:

$$D_{distinguishable}(C_{cover}, S_{stego}) \leq \varepsilon \tag{3}$$

where C_{cover} and S_{stego} represent the cover set and the stego set respectively, ε represents a quantifiable level of security for indistinguishability, ε -security. The three expressions indicate the goal of a steganographic algorithm. We called them the necessary steganographic conditions (NSC).

To facilitate the transmission of secret information, the embedded capacity of the steganographic system [24]

should be high enough. There are already many evaluation criteria for measuring message capacity, such as per-pixel bits, the ratio of secret messages to cover image, and so on.

B. STEGANOGRAPHY SECURITY

Steganography security depends on the means of the attacker. According to Wendy’s work in examining the images, she can be active or passive. When Wendy only checks whether the stego image is natural or normal in the channel transmission, she is called a passive warden. If Wendy could learn from previous attack experience and accumulate knowledge about the stego-system, she is an active warden. She could try to slightly modify the communicated objects or detect the existence of covert communication by extracting secret messages directly. Many reviews of steganography focus on the passive warden mode. In practice, it is common for Wendy to have both active and passive responsibilities as a warden. According to the Kerchhoffs’s principle [25] of security systems, Wendy has complete knowledge of the steganographic algorithm that Alice and Bob might use.

Active attack: In the case of an active warden, steganographic security is mainly concerned with the difficulty of message extraction. The traditional realization of steganography that lacks shared secrets is through obscure security forms. Hopper [26] and Katzenbeisser and Petitcolas [27] independently proposed the complexity theory definition of steganographic security. In our recent work [28], a stego-security classification is proposed based on the four levels of steganalysis attacks:

a) Stego-Cover Only Attack (SCOA): In this case, we assume that the steganalysis attacker can only access to a set of stego-covers.

b) Known Cover Attack (KCA): In this case, being able to perform SCOA, the attacker can also obtain some cover images and their corresponding stego images., the number of pairs is limited within polynomial complexity.

c) Chosen Cover Attack (CCA): In this situation, an attacker can use the steganographic algorithm to perform multiple message embedding and extraction operations with a priori knowledge under KCA., The number of invocation operations is limited within polynomial complexity.

d) Adaptive Chosen Cover Attack (ACCA): The ACCA mode means that when the CCA mode challenge fails, another CCA attack can be performed until the attack is successful.

Under this definition, the steganalyzer does not need to know the probability distribution of the cover, but only assumes that Wendy can access to a black box to generate the cover. She can sample the cover from the black box. Meanwhile, steganographic security is established through the adversarial game between warden and judges. This method is based on the classification standard of security level in cryptography. However, the difficulty of constructing this black box limits the development of security based on computational complexity in the case of active attacks. Fortunately, the generative model provides a technique for

building this black box. Security evaluation criteria based on complexity theory will play a more significant role in the evaluation of steganographic security.

Passive attack: The key issue of steganography security is the indistinguishability between the stego image and cover image. The indistinguishability includes the imperceptibility for the human visual and machine statistic analysis system. Therefore, we have

$$D_{\text{distinguishable}}(C_{\text{Cover}}, S_{\text{Stego}}) = D_{\text{visual}}(C_{\text{Cover}}, S_{\text{Stego}}) + D_{\text{statistical}}(p_{\text{cover}}, p_{\text{stego}}) \quad (4)$$

where $D_{\text{visual}}(C_{\text{Cover}}, S_{\text{Stego}})$ denotes the perceptibility by human, and $D_{\text{statistical}}(p_{\text{cover}}, p_{\text{stego}})$ indicates the statistical distance between the distribution of cover images and the distribution of stego images. In terms of human vision, most current steganography methods can achieve indistinguishable between the stego image and the cover image, which can be represented as $D_{\text{visual}}(C_{\text{Cover}}, S_{\text{Stego}}) = 0$. Statistical indistinguishability is the most studied area of steganographic security. Cachin [29] defined quantified security for a steganography scheme by the relative entropy between the cover distribution p_{cover} , and stego distribution p_{stego} :

$$D_{\text{statistical}}(p_{\text{cover}}, p_{\text{stego}}) = D_{\text{KL}}(p_{\text{cover}} || p_{\text{stego}}) = E_{p_{\text{cover}}}[\log \frac{p_{\text{cover}}}{p_{\text{stego}}}] \quad (5)$$

Based on this definition, if $D_{\text{KL}}(p_{\text{cover}} || p_{\text{stego}}) \leq \epsilon$, the steganography system is called ϵ -security. If $\epsilon = 0$, the scheme is called perfectly secure. Although the definition of security based on information theory is popular, it is an ideal way to define security regardless of its implementation. It requires the assumption that Wendy fully understands the probability distribution of the cover and stego sets.

At the same time, there are other ways to define steganographic security. ROC performance is adopted as an alternative security measure [30]. Steganographic security is defined with a functional performance by the steganalysis tools. The maximum mean discrepancy (MMD) [31] is also be considered as a measure of steganographic security. The advantage of the method is numerically stable, even in high-dimensional space.

C. STRATEGY IMPLEMENTATION

In this paper, the embedding algorithm associates every message m with a pair $[s, \pi]$, where s is stego image, and π is the probability distribution for a specific embedding operation, $\pi(s) = P(S = s | m)$. Unlike the [32], in this paper, we do not give the cover image c explicitly and treat image steganography as a mapping process from message m to stego image s .

If Bob receives s , Alice could send up to

$$H(\pi) = \sum_{s \in S} \pi(s) \log \pi(s) \quad (6)$$

bits message on average. In this situation, the average distinguishability can also be denoted by:

$$E_{\pi}[D_{\text{distinguishability}}] = \sum_{s \in S} \pi(s) D_{\text{distinguishability}}(s) \quad (7)$$

where $D_{distinguishability}(s)$ is a metric indicating that the cover image s is indistinguishable from the natural cover image c . Similar to [32], the task of embedding can assume two forms:

Distinguishability-limited sender (DLS): In this mode, the average payload will be maximized given fixed indistinguishability:

$$\operatorname{argmax}_{\pi} H(\pi) \text{ st. } E_{\pi}[D_{distinguishability}] \quad (8)$$

Although DLS corresponds to a more intuitive use of steganography since images with different level of noise, it is rarely used in practice. If we can find a procedure to create a stego image with a fixed average distinguishability, maximize the average payload will be the core aim. In the next section, we will see that the cover synthesis steganography algorithm can be realized with this form.

Payload-limited sender: In this mode, the indistinguishability metric is minimized, given the size of the transmitted message.

$$\operatorname{argmin}_{\pi} E_{\pi}[D_{distinguishability}] \text{ st. } H(\pi) = m \quad (9)$$

The Payload-limited sender is commonly used in steganography by cover modification, in which $D_{distinguishability}$ is often replaced by a defined distortion function. Minimizing $D_{distinguishability}$ indicates modification operation introduces the least abnormalities in stego image.

A practical steganographic scheme can be divided into three different fundamental architectures according to the different ways of obtaining the stego image.

1) COVER MODIFICATION

There are mainly two types of approaches for steganography by cover modification. One kind is to maintain the invariance of a statistical model [31], and the other type of methods implement embedding by minimizing a specific distortion function. [32], as shown in Fig. 2.

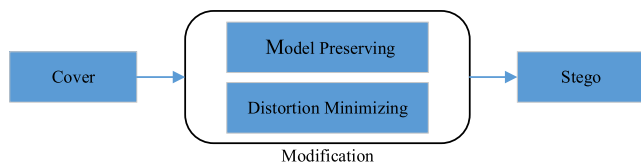


FIGURE 2. Steganography with cover modification.

Cover modification strategy that maintains a specific statistical model is not safe enough in the face of well-designed steganographic features [33], [34]. Steganography, based on minimizing distortion, is more straightforward and attractive. It abandoned the need for statistical modeling of the cover source and instead sought to reduce the distortion [32], [35], [36] introduced by the embedding. The method based on minimizing distortion is state-of-art in steganography with cover modification. This method has a high embedding capacity and is convenient and straightforward to implement. The distortion function is usually a simple

additive distortion. Some improved distortion functions are also proposed [37]–[39].

However, the definition of distortion is too vague to detect the differences introduced into a natural image by a modification accurately. Furthermore, stego s is highly correlated with specific cover c , a well-trained classifier that training on data set C_{cover} and S_{stego} can perform steganalysis. Methods of cover modification always assume that the modification can avoid the attention of human vision, $D_{visual}(C_{cover}, S_{stego}) \approx 0$. However, modification inevitably leads to the difference between the cover distribution and the stego distribution. $D_{statistical}(p_{cover}, p_{stego}) \neq 0$. In the case of passive attacks, the relationship between distortion $D_{distortion}$ and statistical distinguishability $D_{statistical}$ is far from clear.

2) COVER SELECTION

Cover selection methods can be divided into two ways. One is to select a candidate image for modification [40]–[42]. These methods look for a suitable cover in the database to implement the steganography by cover modification. Although these methods are called cover selection, they are still essentially a cover modification method. We do not treat these schemes as cover selection steganography. The other is to select a cover image as a stego image without modification, as shown in Fig. 3.

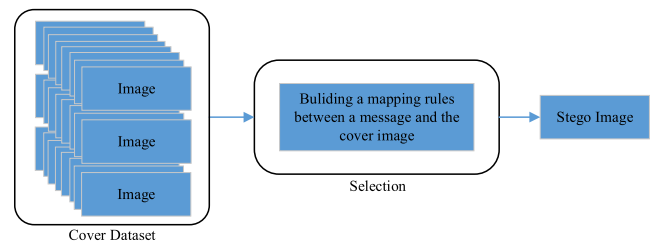


FIGURE 3. Steganography with cover selection.

The essence of the cover selection method is to establish the mapping rules between a message and a stego image. Zhou et al. [43] introduce a cover selection steganography scheme by using the bag of words model [44] (BOW) in computer vision. Firstly, visual words from an image set are extracted using a BOW model. Then a mapping relationship between keywords in the message and visual words in the image is established. According to the known message and a set of rules, the selection method looks for the image that can extract the message as a cover image in the image dataset. This set of rules is essentially a secret key for cover selection steganography. However, as the mapping relationship between message and stego is fixed and the mapping structure is usually quite simple, it is easier to deduce the mapping rules between message and stego through some observations under the active attack. Another problem is that this simple mapping rule leads to low embedding rates, which hinders the deployment of such algorithms in practical applications.

3) COVER SYNTHESIS

The third strategy is based on image synthesis. In this method, Alice tries to create a new image to carry the required secret information. If the synthesis image is real enough that $D_{indistinguishable}(C_{cover}, S_{stego}) = 0$, then a secure steganographic system can be achieved theoretically. At present, there are two kinds of steganographic methods with image synthesis, as shown in Fig. 4.

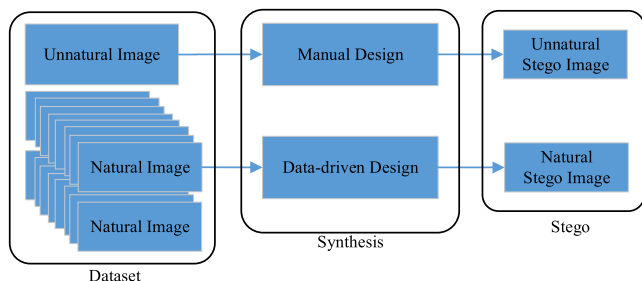


FIGURE 4. Steganography with cover synthesis.

Since the realistic image synthesis is a difficult problem, the traditional cover synthesis method tried to achieve steganography tasks via unnatural image synthesis, texture image [45], and fingerprint image [46]. Otori and Kuriyama [47], [48] first try to combining information hiding with pixel-based texture synthesis. Wu and Wang [49] proposed a reversible texture image synthesis for steganography. Qian *et al.* [50] propose a steganography method in which secret messages are hidden in a texture image during the process of synthesizing. [51], [52] introduce a deformation-based texture for information hiding. Li and Zhang *et al.* [53] propose a construction-based steganography scheme which conceals a secret message into a fingerprint image. The premise of texture synthesis for steganography is to assume the stego object can be an image without semantic information. Meaningless image limits the application of texture synthesis steganography in a larger field.

The other approach is to train a generator by a generative model with a large amount of data. Stego images can be obtained from the realistic image generator. A probability distribution described by the generative model is p_{model} or p_g . In some cases, the model estimates p_{model} explicitly. Furthermore, if the images obtained by the generative model are treated as the stego images, the distribution of the generated samples can also be denoted by p_{stego} . The maximum likelihood estimation used for estimating parameters of the density function is computationally tractable. While variational methods and sampling methods such as Markov chain Monte Carlo are used for intractable density function, requiring the use of approximations to maximize the likelihood. Because of the complexity and high dimension of natural images, it is impossible using an explicit density function to describe the distribution of natural images. Fortunately, the GAN model uses an indirect method to

obtain the distribution of real images, which does this by generating samples rather than estimating the specific form of the distribution. In Section VI, we will see how to design the steganography method using an image generator, which is obtained by training the GAN model. Some other generative models can be used for designing steganography algorithms. Chen *et al.* [124] discussed how to apply the VAE [122], and flow-based generative models [123] in steganography. Our paper mainly analyzes the application of GAN in steganography.

D. SUMMARY ON TRADITIONAL STEGANOGRAPHY

Under passive attack, a steganographer tries to find an algorithm satisfying steganography condition $p_{stego} = p_{cover}$. For technical feasibility for steganographic security, Fridrich [1] ignores the fundamental question of whether it is feasible to assume that the distributions p_{cover} and p_{stego} can be estimated in practice or even whether they are appropriate descriptions of the cover-image source. Therefore, the choice of the specific form of p_{cover} becomes the critical issue in designing of steganography method. Traditional steganography research has focused on methods based on the cover modification.

Most of the methods based on modification try to ensure that the modification operation should keep the invariance of a specific statistical characteristic, that is $p_{cover} \neq P_{cover_specific}, P_{cover_specific} = P_{stego_specific}$. The disadvantage is that the opponent can usually identify the statistic beyond the selected model reasonably easily, which allows the reliable detection of embedded changes.

In the steganographic scheme of cover synthesis, the distribution of the stego images p_{stego} should be close enough to the real distribution of the cover image p_{real} . Although the actual image of distribution can not be given explicitly, we can approximate the real image distribution p_{real} by describing the distribution of existing data, $p_{real} \approx p_{data}$. As discussed in the previous section, GAN allows us to train a generator with an adversarial learning model. The distribution of the samples sampled from the generator satisfies $p_g = p_{data}$. When we get the stego image directly from the generator, we can achieve statistical indistinguishability. When a proper description of the cover images is obtained, both the indistinguishability security and the Kerckhoffs's principle require further attention. To understand the characteristics of GAN-based steganography methods clearly, the following Section II will briefly discuss the basic principles and features of GAN. As we will see, the underlying fundamental questions which are neglected by traditional steganography is what GAN wants to solve.

III. GAN PRELIMINARIES

A. CORE CONCEPTS

The basic idea of GANs is an adversarial game between two players, as illustrated in Fig. 5. The task of generator G is to transform the input noise z into a sample $G(z)$. The discriminator D determines whether the generated fake sample is

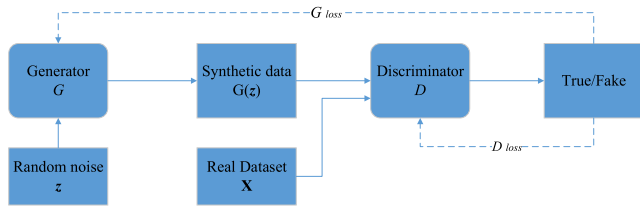


FIGURE 5. The general structure of GAN.

indistinguishable from the real sample. A generative model G is a neural network with parameters θ denoted as $G(\mathbf{z}; \theta)$. The output of the generator can be viewed as a sample from a distribution $G(\mathbf{z}; \theta) \sim p_g$. With a lot of real data \mathbf{x} from p_{data} , the goal of generator training is to make the generator's distribution p_g close to the real data distribution p_{data} .

Goodfellow *et al.* use a multilayer perceptron as a generator. The objective function is shown in Eq. 10:

$$\min \max V(D, G) = E_{\mathbf{x} \sim p_{data}(\mathbf{x})}[\log D(\mathbf{x})] + E_{\mathbf{z} \sim p_z(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))] \quad (10)$$

They also show that the optimization process can be seen as minimizing the Jensen-Shannon divergence (JSD) [3] between real data distribution and generator distribution. More importantly, if both generator and discriminator have adequate capability, the game will converge to its equilibrium with $p_g = p_{data}$. In practice, the parameters for the two networks will be updated in the parameter space.

B. IMPROVEMENTS AND APPLICATION

1) IMPROVEMENTS

The improvements of GAN models can be classified into two aspects: the architecture and the loss function. To be specific, GANs are classified into different types, as shown in Fig. 6.

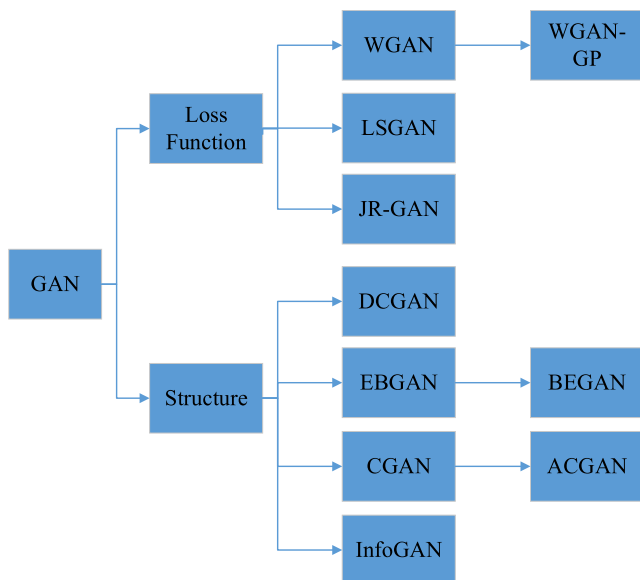


FIGURE 6. Improvements on GAN models.

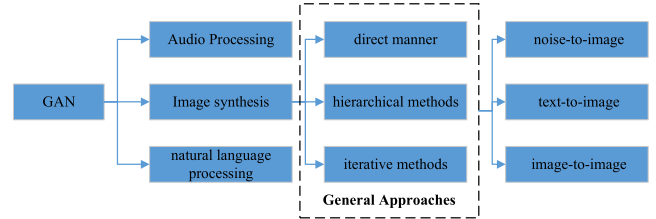


FIGURE 7. Applications with GAN models.

The most famous model is DCGAN. [4], which performs well in image synthesis in the early work of the research., different GAN such as CGAN [54], InfoGAN [8], ACGAN [55] are proposed to control the generated result. Some methods have been proposed for solving the model collapse problem by designing a new loss function such as mini-patch feature [5], MRGAN [56], WGAN [6], and WGAN-GP [57].

2) APPLICATIONS

The application of early GAN was mainly concentrated in the field of computer vision, such as image inpainting [58], captioning [59], [60], detection [61], and segmentation [62]. GAN also has some applications in the field of natural language processing, such as text modeling [10], [63], dialogue generation [64], and machine translation [65].

Huang *et al.* [13] summarize main approaches in image synthesis into three methods, i.e., direct methods, hierarchical methods, and iterative methods. Direct Methods such as GAN, DCGAN, Improved-GAN [5], InfoGAN, f-GAN [66], and GANINT-CLS [67], usually using one generator and one discriminator. The hierarchical approach uses two generators and two discriminators. This approach divides the image into two different pieces of content, such as “style and structure” and “foreground and background.” Hierarchical methods refer to the model, which generates images from coarse to fine using multiple generators with similar or identical structures.

On the other hand, depending on the source of the generated image, image synthesis can also be divided into three different synthesis methods, namely noise-to-image, text-to-image, and image-to-image. Text-to-image synthesis is a research field with excellent prospects. It means that machines can understand the semantic information of the text. GAN provides us with a promising text-to-image synthesis method, such as GAN-INT-CLS [67], GAWWN [68], StackGAN [69], and PPGN [70]. The GAN-based approach so far produces images that are sharper than any other generation method. Image translation is related to style transfer [71], which samples an image with specific content and style by using a content image and a style image. The image-to-image translation by GANs has also been successfully applied in some image or video generation applications [72].

C. STEGANOGRAPHY BY GAN

In this section, we summarize the characteristics of GAN. GAN’s features can be viewed from the following three

aspects: an adversarial game, a generator, or a mapping function. These consistent with the classification of basic steganographic strategies, i.e., cover modification, cover synthesis, and cover selection. The three main types of approaches mentioned in this paper based on the above characteristics are shown in Fig. 8.

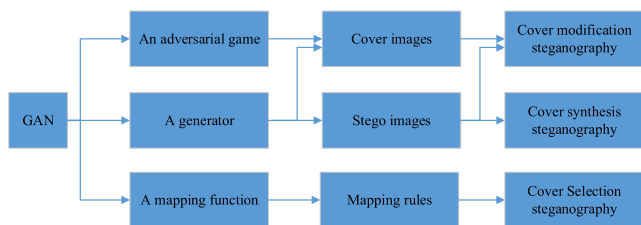


FIGURE 8. The categories for steganography from the point of view of GAN.

Under the first view, GAN is treated as an adversarial game between generator and classifier; both generator and discriminator are equally important. This kind of viewpoint pays attention to the whole process of the adversarial game and pays more attention to the positive effect produced by the discriminator. In fact, there have been some studies on steganography methods based on game theory before the GAN is proposed, such as [73]–[75]. The family by modification-based steganography takes advantage of the concept of a game simulation between two-players: Alice-agent and Eve-agent. Historically MOD [76] and ASO [77] were the algorithms of this type. Recently some researchers take advantage of the adversarial concept by generating a fooling example (see for adversarial example [78]). Still, those approaches are not an adversarial game between generator and discriminator. However, unlike GAN using iterative and dynamic game process, those approaches are not a dynamic process, there is no dynamic adversarial game simulation. They did not attempt to achieve a Nash equilibrium. These methods considered the implications of steganalysis at the beginning of designing steganographic schemes.

On the other hand, this traditional game strategy is more of a theoretical analysis. The steganography based on GAN can be used in creating a practical steganography scheme. The game between steganalyser and steganographer is similar to the generator and discriminator in GAN. Inspired by this similarity, GAN is chosen to improve the performance of the traditional steganography by cover modification. We will discuss the specific process of this part in detail in Section IV.

The second view treats the generator training procedure in GAN as a robust construction method of the mapping function. This mapping function maps a driving signal through a neural network to an image that belongs to a specific image set. It is an interesting idea to apply GAN in the cover selection steganography scheme for building the mapping between message and cover. We will elaborate on the details of this scheme in Section V.

The third view is to regard GAN as a method to construct a powerful generator. As we know, this view treats the result

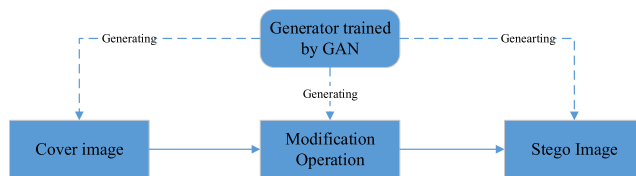


FIGURE 9. The categories for cover modification with GAN.

of the game process, a powerful generator, as the most successful innovation of the GAN model. A much more exciting approach using cover synthesis is to generate stego images by the generator. The critical issue raises how to hide the message in the synthetic image. A typical approach is to obtain a stego image by introducing steganography constraints or a loss term with message extraction. We will discuss some recent researches in detail in Section VI.

IV. COVER MODIFICATION WITH GAN

GAN-based steganography by cover modification (GAN-CMO) focuses on the adversarial game between steganographer and steganalyser. These methods use a generator trained by GAN to construct various core elements in the cover modification scheme. The first strategy is to generate the cover image. The second strategy is to create the modification probability matrix in the framework of minimizing distortion, and the third is to directly use the adversarial game among tripartite, such as Alice Bob and Wendy, to learn a modification algorithm.

A. GENERATING COVER IMAGES

Volkhonskiy et al. [79] proposed the application of GAN to steganography. They construct a special generator for creating cover-image, synthetic images produced by this generator are less susceptible to steganalysis compared to covers. This approach allows for generating more steganalysis-secure cover that can carry messages using standard steganography algorithms such as LSB or STC. They introduce the Steganographic Generative Adversarial Networks called SGAN, which consists of three networks. A generator G , a discriminator D , and a steganalysis classifier S . Classifier S determines if a realistic image is hiding secret information. The workflow of SGAN is illustrated in Fig.10.

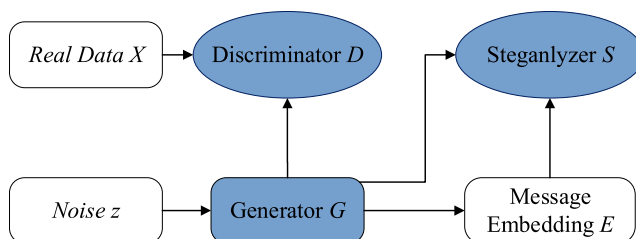


FIGURE 10. SGAN workflow diagram.

SGAN trains G with D and S simultaneously. We can get the game as follow:

$$L = \alpha(E_{x \sim p_d(x)}[\log D(x)] + E_{z \sim p_z(z)}[\log(1 - D(G(z)))] + (1 - \alpha)E_{z \sim p_z(z)}[\log S(\text{Stego}(G(z)))] + \log(1 - S_D(G(z))) \rightarrow \min_G \max_D \max_{S_D} \quad (11)$$

where parameter α $[0; 1]$ denotes the weight between the quality of the generated image against the steganalysis, $S(x)$ is the probability for x is stego image.

Similar to SGAN, Shi *et al.* [17] use the same strategy that generates cover images for steganography with adversarial learning scheme, named SSGAN. The SSGAN also has a generative network and two discriminative networks. Compared with the SGAN, WGAN is proposed for generating higher quality images and improving the training process. A more complex network called GNCNN [80] is chosen as the discriminator D and the steganalyser S .

Wang *et al.* propose another cover image generation method [81], as shown in Fig. 11. Unlike SGAN and SSGAN, a discriminator D determines whether the image is a stego image. $\text{Stego}(G(z))$ and real image x are used as the input for discriminator D . Such improvement make the distribution of the stego image closer to the real data distribution. An interesting result of this scheme is that the images generated directly by the generator may not be realistic, and the fidelity of the stego image is achieved after the modification operation.

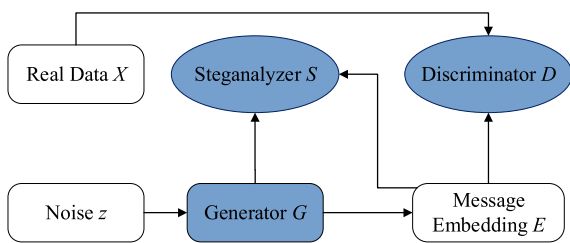


FIGURE 11. Stego-GAN workflow diagram.

B. LEARNING DISTORTIONS

Tang *et al.* [82] proposed an ASDL-GAN model to learn a distortion function automatically. This scheme follows the state-of-art steganography by minimizing an additive distortion function [32]. The change probabilities matrix P can be obtained by minimizing the expectation of the distortion function [83]. The generator G in their scheme is trained to learn the change probabilities P for an input image.

As illustrated in Fig. 12(a), the discriminator D in ASDL-GAN framework adopts the Xu’s model architecture [84]. The embedding simulator (TES), is used as the activation function in the training procedure. The reported experimental results showed that ASDL-GAN could learn steganographic distortions.

It is inspired by ASDL-GAN, UT-SCA-GAN [14] proposed by Yang *et al.* with the same component modules as

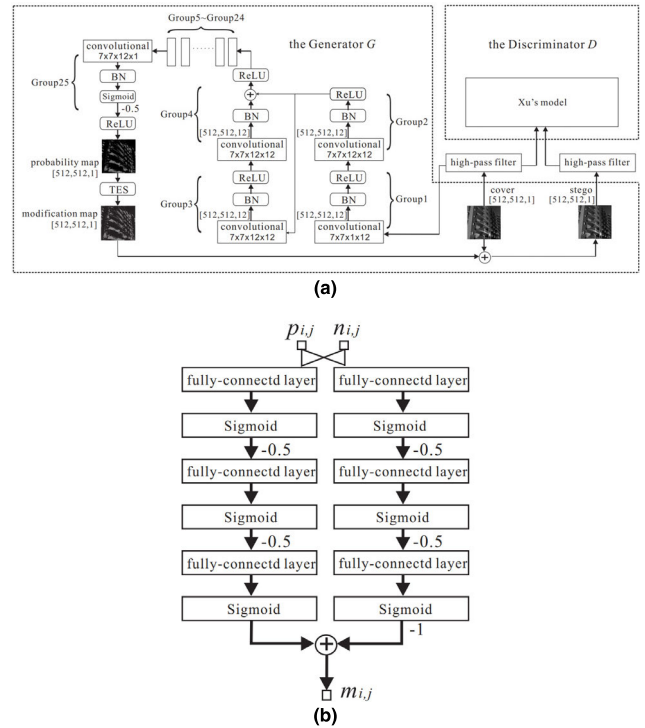


FIGURE 12. (a) Architecture of the ASDL-GAN framework [82]. (b) The structure for TES activation function.

ASDL-GAN: a generator, an embedding simulator, and a discriminator. Compared with the ASDL-GAN, Tanh-simulator, an activation function, is used for the propagating gradient. Besides, a more compact generator based on U-Net [85] has been proposed. The experimental results show that this framework can improve security performance. At present, there is no guarantee [86] that the probability map obtained will defeat the security performance of HILL or S-UNIWARD with STC in practice. It is also unclear whether the loss of the generator must incorporate terms related to safety and terms of payload size.

C. EMBEDDING AS ADVERSARIAL SAMPLES

Some researchers have also designed steganography with the idea of adversarial examples [87]. However, merely adding perturbations directly to a stego image can also result in instability of message extraction. Tang *et al.* [88] proposed an adversarial embedding (ADV-EMB) method, which tries to modify the cover image for message hiding while fooling a steganalysis classifier. The ADV-EMB scheme is illustrated in Fig. 13.

The pixels of candidate stego image is divided into two groups, one group of pixels is used for modification-based embedding, and a tunable group of pixels is used for perturbation as an adversarial sample to resist steganalysis. ADV-EMB adjusts the cost of modification operation with back-propagation on the gradient of the steganographic analyzer. Their experiments show that ADV-EMB achieves better security performance.

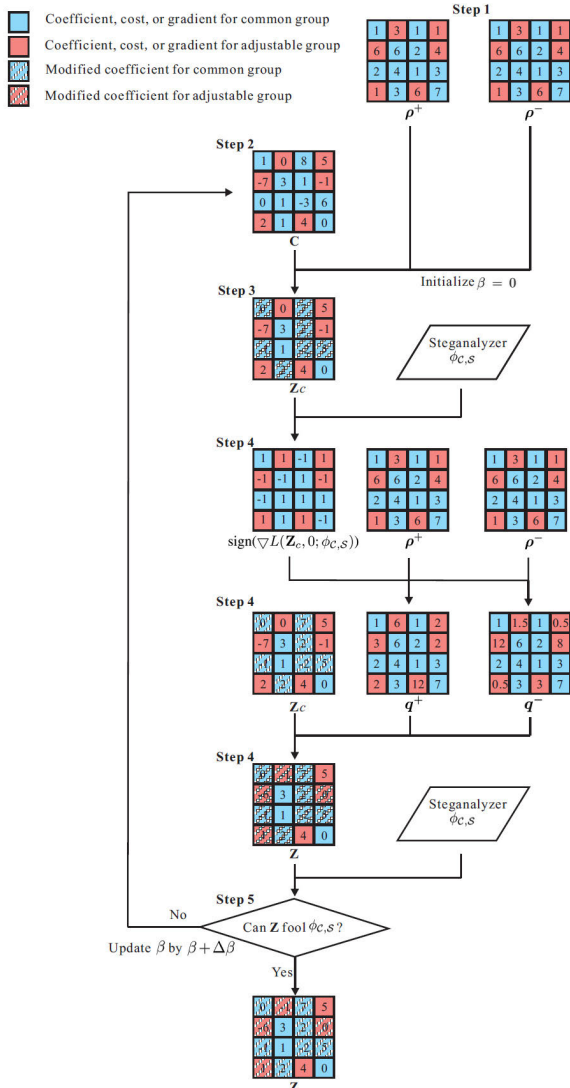


FIGURE 13. The model architecture of ADV EMB scheme [88].

Similar to [88], Ma et al. [89] modify the image pixel following the adversarial gradient map while embedding. The adversarial gradient map is the matrix generated from the neural network model and has the same size as the cover image. Each element of the adversarial gradient map is the gradient value that make the steganalyzer tend to have a false classifying result.

D. SUMMARY ON COVER MODIFICATION

SSGAN [17] construct a special cover-image generator; they can use standard steganography algorithms such as LSB or for information hiding. [82] and [14] train a generator of modification probabilities matrix for minimizing a suitably defined additive distortion function. [15], [78], [88], [89] learn a whole cover modification steganographic algorithm using GAN. They focus on the adversarial game between steganography and steganalysis. They both introduce a steganalyzer against the steganography either explicitly or implicitly.

Although these methods have achieved better anti-analysis capability than traditional steganography methods, these methods are still faced with traditional security threats when Wendy can get all the information on the algorithm to obtain stego and cover. In theory, these methods can't resist more powerful steganalysis tools, since the embedded operation will inevitably cause some abnormal changes.

V. COVER SELECTION WITH GAN

GAN-based steganography by cover selection (GAN-CSE) aims to establish the mapping relationship between message and cover. As far as we know, there is very little literature on this subject that attempts to use GAN to design the steganography scheme, Ke et al. [16] made a preliminary attempt on this subject.

A. COVER FIRST GENERATIVE STEGANOGRAPHY

Ke et al. [16] proposed generative steganography by the cover selection that meets Kerckhoffs' principle (GSK). The idea is that the sender establishes a mapping relationship using a generator between the message and the selected image. For the receiver, the message is directly generated by the selected image. In [28], this method is also called cover first generative steganography (CFGs). The essence of this method is to establish a mapping relationship between the cover image and secret message so that a cover image will naturally turn to be a stego image. The statistical steganographic analysis does not work because there is no operation for cover modification. Ke et al. establish a mapping relation between message and cover image using GAN. To ensure the security, Kerckhoffs' principle is also introduced in their GSK method. Fig. 14 shows the three message extraction scenarios under this framework.

As for the receivers (Fig.14c): Case 1, the only k is received corresponding to a failed message extraction, only

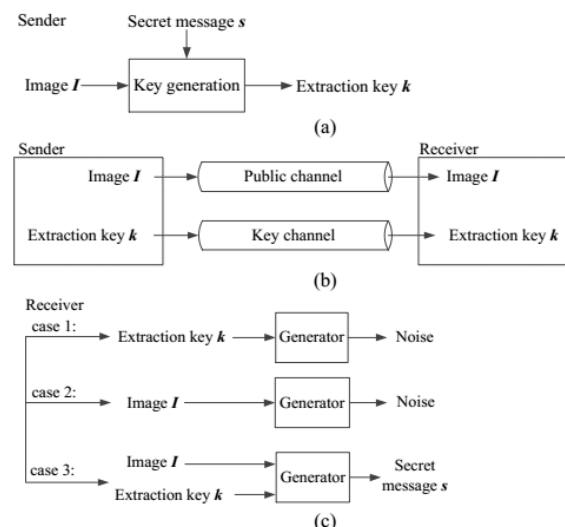


FIGURE 14. The model architecture of GSK method [16].

noise can be recovered. *Case 2*, only I is received corresponding to an intercept from attackers. There is also noise output for the attacker. *Case 3*, when I and k are both obtained, the message s could be recovered. Two mapping relationships between the key k and the message s and the relationship between the cover I and the message s are constructed by a Message-GAN and Cover-GAN, respectively. Message-GAN, which implement by InfoGAN [8], is to use feature codes to control the output. Cover-GAN, which is similar to Abadi and Andersen [90] method for cryptography, is used to determine the generation of the message s .

B. SUMMARY ON COVER SELECTION

There are few studies on cover selection steganography based on generative models. This type of approach treats the generator as a mapping between a message and an existing natural image. The advantage of [16] is that the image is 100% natural due to no modification. For the moment, the low embedding capacity of cover selection steganography is still a bottleneck to its development.

VI. COVER SYNTHESIS WITH GAN

GAN-based steganography by cover synthesis (GAN-CSY) usually creates stego images by generator trained by GAN. In our opinion, the key to steganography by image synthesis is that the stego image should be obtained directly from a black box, such as a generator. Since the most significant advantage of GAN is the ability to generate realistic natural images, we will see in this section how to use generators to create stego images.

A. SUPERVISED STEGO IMAGE SYNTHESIS

Similar to Abadi and Andersen [90], Hayes and Danezis [15] try to use a neural network to learn a steganography algorithm with adversarial training. In their framework, the three players, Alice, Bob, and Eve, are neural networks. $\theta_A, \theta_B, \theta_C$ denote the parameters for the networks, respectively. The full scheme is depicted in Fig.15.

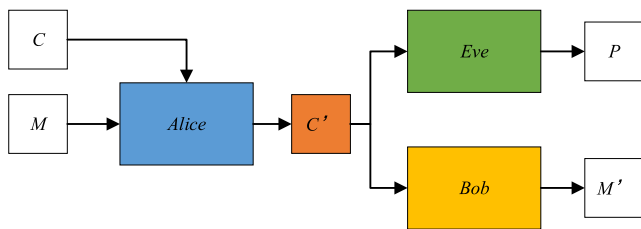


FIGURE 15. 3-PLAYERS GAME for steganography by GAN [15].

In Fig.15, Alice uses a cover image, C , and a secret message, M to generate a stego image C' , Bob tries to recover the message M' from C' . Eve outputs a probability P to indicate the likelihood of a secret message in the image. Alice hopes to learn a steganography scheme in which Eve outputs $P = 1/2$. $A(\theta_A, C, M)B(\theta_b, C')$ and $E(\theta_E, C, C')$ are output for Alice, Bob, and Eve, respectively. To design a steganographic

algorithm, three loss-functions L_A, L_B , and L_E are given as the loss of Alice, Bob, and Eve.

$$L_B(\theta_A, \theta_B, M, C) = d(M, B(\theta_B, C')) + d(M, B(\theta_B, A(\theta_A, C', M))) + d(M, M') \tag{12}$$

$$L_E(\theta_A, \theta_E, C, C') = -y \log(E(\theta_E, x)) - (1 - y) \log(1 - E(\theta_E, x)) \tag{13}$$

$$L_A(\theta_A, C, M) = \lambda_A d(C, C') + \lambda_B L_B + \lambda_E L_E \tag{14}$$

where $y = 0$ if $x = C'$ and $y = 1$ if $x = C$, $d(C, C')$ is the distance between the C and C' , and hyperparameters $\lambda_A, \lambda_B, \lambda_E \in \mathbb{R}$ define the weight for each loss term.

Zhang *et al.* [91] propose an end-to-end model, called STEGANOGAN, for image steganography. They use adversarial training to solve the steganography task and treat the messages embedding and extraction as encoding and decoding problems, respectively. The architecture of STEGANOGAN consists of three sub-modules, as shown in Fig. 16, the image encoder uses the cover image and the message to generate a stego image; a decoder is going to recover the message with a stego image, and an auxiliary Critic network evaluates the quality of the stego image.

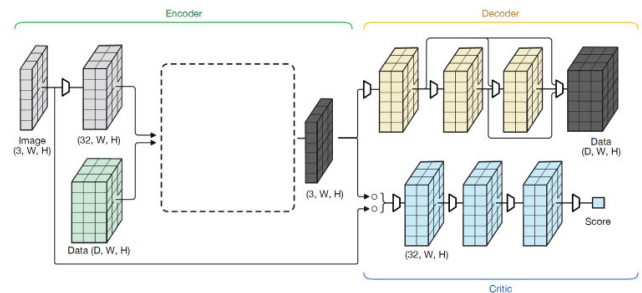


FIGURE 16. The architecture for STEGANOGAN model [91].

The training process is divided into two parts. Three losses: the cross-entropy loss L_d for message decoding accuracy, the similarity loss L_s between stego and cover, and the fidelity loss L_f of the stego image using the critic network. The training objective is to

$$\text{minimize } L_d + L_s + L_f. \tag{15}$$

They minimize the Wasserstein loss to train the critic network.

In [92], Zhu *et al.* also trained encoder and decoder networks to implement message embedding and extraction. They introduce various types of noise between encoding and decoding to increasing robustness but focuses only on the set of corruptions that would occur through digital image manipulations. Similar to [92], Tancik and Ren [93] achieve robust decoding even under “physical transmission” by adding a set of differential image corruptions between the encoder and decoder that successfully approximate the space of distortions.

Although the above algorithms generate stego images through neural networks, it should be emphasized that the ideas of these methods are substantially dependent on a specific cover image, and we call it the *steganography by supervised cover synthesis* (SSCS). The stego image generated by a neural network is highly correlated with the cover image, so those algorithms are similar to the steganography by cover modification.

B. UNSUPERVISED STEGO IMAGE SYNTHESIS

1) STEGANOGRAPHY WITHOUT EMBEDDING

Hu *et al.* [19] proposed a stego-image synthesis method without embedding (SWE). In our opinion, a more appropriate term would be “steganography without modification”, since embedding operations should be a general term for information hiding operations and should include modification, selection, and synthesis. In their method, the secret messages are mapped into a noise vector is sent to the generator as input to produce a stego image. In Hu’s method [19], stego images are generated with the noise in an unsupervised manner. We call this the *steganography by unsupervised cover synthesis* (SUCS). The proposed SWE framework consists of three phases, as illustrated in Fig. 17.

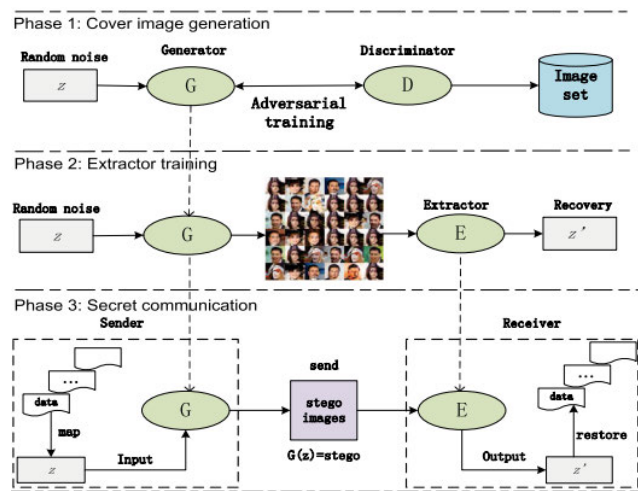


FIGURE 17. The framework of SWE method [19].

The generator G is trained with a dataset in the first phase. After this phase, the generator that can create realistic fake images is obtained. During the second phase, an extractor E is trained by a message extraction loss function. The goal of this phase is to recover the message from the generated stego image. The loss of extractor training is illustrated as follow:

$$\begin{aligned}
 L(E) &= \sum_{i=1}^n (z - E(stego))^2 \\
 &= \sum_{i=1}^n (z - E(G(z)))^2 \quad (16)
 \end{aligned}$$

In the secret communication phase, the sender builds a relationship between noise and message, in their scheme, both secret message m and vectors z are segmented for mapping. The receiver can use E to recover noise vector z , and then the secret message is obtained by the mapping relationship.

The highlight of this paper is training a special extractor for noise (message) extraction.

2) STEGANOGRAPHY BY WGAN-GP

Inspired by Hu’s method, Li *et al.* [94] propose a new framework that trains the message extractor and stego image generator at the same time. WGAN-GP instead of DCGAN is adapted to generate a stego image with higher visual quality. In their method, Generator G is trained in a mini-max game to compete against the Discriminator (D) and Extractor (E), as illustrated in Fig.18. The objective function for training this model is as follows:

$$\begin{aligned}
 \min \max \min J(D, G, E) &= \{E_{x \sim p_{data}(x)}[D(x)] - E_{z \sim p_z(z)}[DG(z)] \\
 &+ \lambda E_{\hat{x} \sim p_{data}(\hat{x})}[\|\nabla_{\hat{x}} \|D(\hat{x})\|_2 - 1]\} \\
 &+ \beta \{E_{z \sim p_z(\hat{x})}[\log(z - E(G(z)))]\} \quad (17)
 \end{aligned}$$

where β is a positive number that balances the importance of realistic images and correct extraction rate of noise z . The second term on the right-hand side is the objective function of WGAN-GP [57]. The parameter λ is the gradient penalty coefficient.

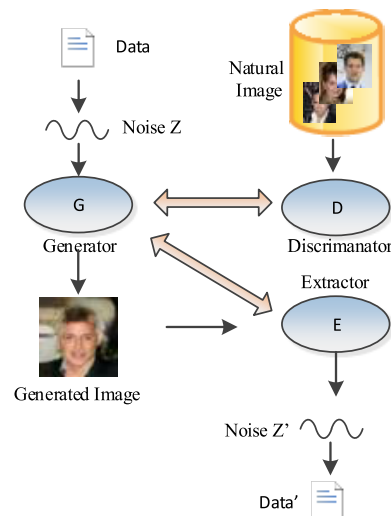


FIGURE 18. The framework of the WGAN-GP [94].

C. SEMI-SUPERVISED STEGO IMAGE SYNTHESIS

1) STEGANOGRAPHY BY ACGAN

To allow for semi-supervised learning a steganographic scheme, we can add the task-specific auxiliary network in the original GAN. Inspired by ACGAN, Liu *et al.* [95] first proposed a stego-image generation method by ACGAN. This method establishes a mapping relationship between the class labels of the generated images and the secret information, both class labels and noise put into the generator for stego image generation directly. We call it the *steganography by semi-supervised cover synthesis* (Semi-SCS). The receiver extracts the secret information from the hidden image through a discriminator.

The ACGAN-based cover synthesis method attempts to establish a correspondence between image categories and secret information. ACGAN for generating the stego image, as illustrated in Fig.19.

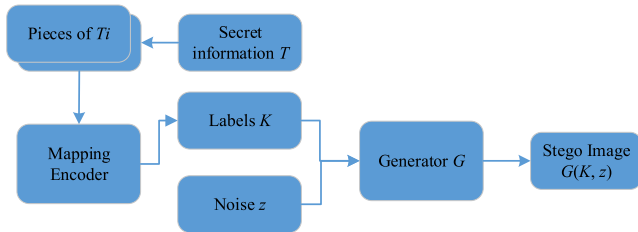


FIGURE 19. The framework of steganography with ACGAN.

At the message extraction phase, the stego image is fed into a discriminator for getting the pieces of secret information. Hu’s method [19] and Li’s [94] methods are necessarily the same as Liu’s method [95], all of which attempt to create a mapping between the input vector of the generator and the secret message. The former establishes the mapping between noise z and the message, while the latter utilizes the auxiliary control information, such as labels.

2) STEGANOGRAPHY BY CONSTRAINT SAMPLING

Liu *et al.* [18], [96] proposed generative steganography by sampling (GSS). In this scenario, the steganographic embedding operation becomes an image sampling problem. They treated stego image generation as an optimization problem of minimizing the distribution distance between the data image and the cover image:

$$Gen(\mathbf{m}, \mathbf{k}) = \arg \min_{\mathbf{y} \sim p_{stego}} D_{JS}(p_{stego} \cdot p_{data}) \quad (18)$$

$$st.Ext(\mathbf{y}, \mathbf{k}) = C_k \mathbf{y}, \quad (19)$$

where $Gen(\cdot)$ is an image generator, and C_k is the secret key \mathbf{k} . The stego image, \mathbf{y} , does not depend on any specific cover, which follows the distribution $p_g, \mathbf{y} = G(\mathbf{z})$.

To implement this solution, they train an image generator by DCGAN, as illustrated in Fig. 20(a). The goal of training is to be able to get realistic fake images. Ideally, it reaches an equilibrium state, $p_{stego} = p_{data}$.

Then, constrained sampling of the image is achieved by defining a message extraction loss constraint, as shown in Fig. 20(b). More specifically, The process of finding a cover image \mathbf{y} can be regarded as an optimization problem as follows:

$$\hat{\mathbf{z}} = \arg \min_z (L_m(\mathbf{z}|\mathbf{m}, \mathbf{k}) + \lambda L_p(\mathbf{z})) \quad (20)$$

where $\hat{\mathbf{z}}$ is the “closest” encoding of stego image, and L_m and L_p denote the message loss and the prior loss. Back-propagation to the input noise \mathbf{z} is introduced for solving this optimization problem. Under the guidance of this framework, they implemented a digital Carden grille steganography scheme using image completion technology [58].

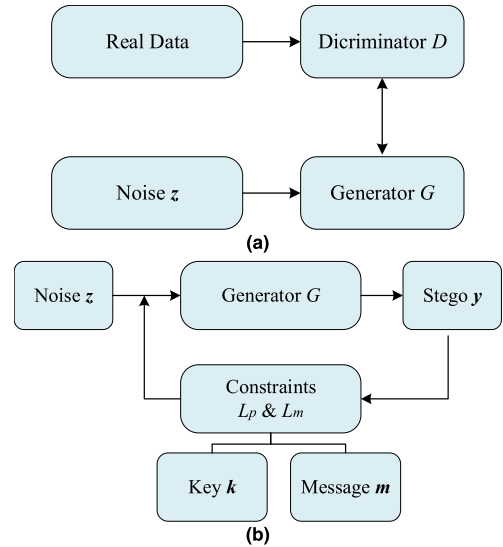


FIGURE 20. Workflow for GSS (a) Training a image generator; (b) finding a stego image with constraints.

In this scheme, the image completion technology makes the scheme closer to the idea of Cardan grille. At the same time, the method becomes a semi-supervised cover synthesis (semi-SCS) method. Image completion technology is not necessary, that is to say, this scheme can be converted into an unsupervised manner, and extended to more image synthesis applications. In this framework, cover synthesis becomes an optimization problem that satisfies both message loss constraint and image perceptual constraint. Unlike Hu’s method [19], the GSS framework provides an alternative way to cover synthesis using generators. In Hu’s paper, after the message-noise map and well-trained extractor are ready, the cover image can be obtained by using noise once. In contrast, in GSS scheme [96], the stego image is achieved via an iterative sampling method step by step.

3) STEGANOGRAPHY BY CYCLE GAN

In addition to using noise, labels, and corrupted images to generate stego images, some researchers treat the cover synthesis as an image-to-image translation problem. The image-to-image translation is a transformation that converts one type of image to another. A very famous model for image translation is CycleGAN [97]. Although CycleGAN lacks the supervision of the pairing example form, it can take advantage of the supervision at the collection level. CycleGAN is able to convert an image from class X to a class Y by a transform F . It can also convert it back to class X by transforming G . CycleGAN trains transforms F and G by minimizing the adversarial loss L_{GAN} and cycle consistency loss L_{cyc} . Chu *et al.* [98] first claim that CycleGAN can be viewed as an encoding process for information hiding. By treating CycleGAN’s training process as a generator of training adversarial examples and demonstrating that cyclical consistency losses cause CycleGAN to be particularly vulnerable to adversarial attacks.

Since CycleGAN’s image transformations have some reversible properties, Di et al. [99] proposed a cover synthesis scheme by cycleGAN with reversible properties. Inspired by Hu’s framework, they introduced cycleGAN into the new framework and used it for the reversible recovery of the cover image, which is generated by a noise vector. Similar to [100], the transformed image can also be regarded as a special encrypted image. In addition, a new extractor is trained to extract the secret data, which also makes the data hiding framework reversible. The illustration of Di’s method has been shown in Fig. 21.

We also regard image synthesis steganography as *generative steganography*. It refers to the means of directly obtaining a stego image by a generator without a specific cover image.

1) SENDER MODE

With GAN’s generator, realistic images are sampled from the distribution of a dataset. Sampling a stego image from a generator makes the steganography problem a sampling process. Steganography, by cover synthesis, also has two implementation strategies.

Payload-limited

Sender: In practice, it is difficult to achieve the optimal, which satisfies $p_g = p_{data}$. In the case that the message length is limited to m bits, the cover synthesis can be regarded to minimize the distance between p_{stego} and p_{data} :

$$Emb(m, k) = \underset{y \sim P_{stego}}{\operatorname{argmin}} D(p_{stego}, p_{data}) \tag{21}$$

$$\text{st. Ext}(Emb(m, k), k) = m, \forall m \in \{0, 1\}^m \tag{22}$$

Distance-limited Sender: Due to the randomness of the images generated by the generator, when the distance between the distribution of the cover image and the real data distribution is within an acceptable range, the steganography by cover synthesis can also be regarded as an optimization problem to maximize the capacity of the message:

$$Emb(m, k) = \underset{y \sim P_{stego}}{\operatorname{argmax}} |\text{Ext}(Emb(m, k), k)|_m \tag{23}$$

$$\text{st. } D(p_{stego}, p_{data}) < \varepsilon \tag{24}$$

where $|\cdot|_m$ denotes the length of message m . The goal of steganography by cover synthesis is to increase the capacity under the premise of satisfying a metric.

The distance-limited sender by cover synthesis has three differences compared to the mode adopted by the minimizing distortion. First, this scheme directly minimizes the distribution distance rather than the distortion caused by modifying operation. Second, the scheme is straightforward to introduce a secret key, making the scheme meet the Kerckhoffs’ principle. Third, the algorithm by minimizing distortion usually adopts the Payload-limited sender method to design the steganographic scheme. The more intuitive use of steganography should be the Distance-limited Sender mode. The distance-limited mode is similar to the mode with distortion-limited, but there is a fundamental difference in steganography security. The relationship between distortion and steganography security is ambiguous. The process of training a generative model is theoretically reducing the distribution distance, which makes the distance-limited mode more perceptive. All of these methods, including the ACGAN-based method [95], SWE method [19], [94], cycleGAN-based method [99], and GSS method [96], adopt the distance-limited sender mode. They introduce message-noise mapping or message loss constraints after the generator is trained. The trained generator represents that a

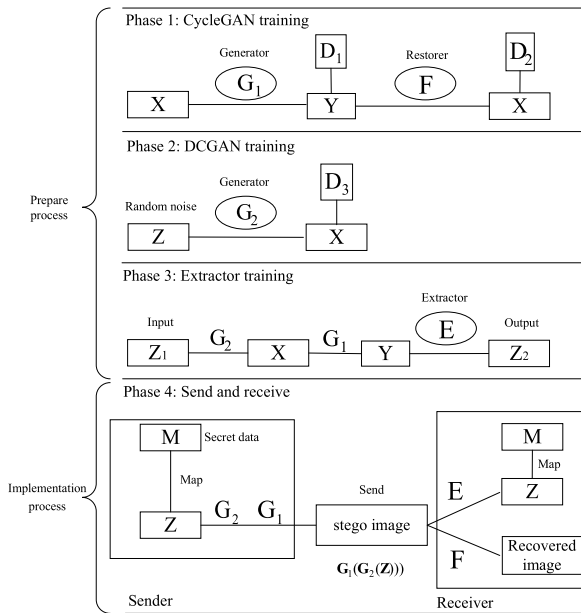


FIGURE 21. The workflow of the method [99].

In phase 1, a generator G_1 and a restorer F are generated by CycleGAN. With two discriminators D_1 and D_2 , two transformations achieved $G_1: X \rightarrow Y$ and $F: Y \rightarrow X$, where X and Y are image collections. In Phase 2, a generator G_2 is generated by the DCGAN method with the help of discriminator D_3 . In Phase 3, based on the two discriminators G_1 and G_2 , we can get the transformation from random noise to stego image set Y . Then, a new extractor E is trained with a neural network, which ensures that the generated output Z_2 is the same as the input Z_1 as closely as possible. Before data hiding, the sender sends extractor E and restorer F to the receiver. Both sides learn a mapping from secret data M to noise Z . The image generated by G_1 and G_2 can be regarded as a cover image and marked image. Then, the sender sends the marked image $G_1 G_2(Z)$ to the receiver. At the receiver side, a recover image can be obtained, and the embedded data can be extracted.

D. SUMMARY ON STEGO IMAGE SYNTHESIS

Although there’s not a lot of literatures on generating stego images with generators, some of them are attractive and representative. In this section, we will further analyze the characteristics of these methods and summarize some general rules.

fixed distribution distance, and the message mapping or message loss constraint aim to improve the embedding capacity.

2) MESSAGE EMBEDDING AND EXTRACTION

The goal of cover synthesis is to generate realistic images while hiding messages. Traditional GANs focus on finishing the realistic image generation task. For steganographers, the most critical mission is how to embed and extract messages correctly. Interestingly, we will see that, in contrast to traditional steganography schemes, which focus on the design of embedding operation, including modification and selection. In the steganography by cover synthesis based on the generator, message extraction and embedding procedures combine in an integral whole. In some circumstances, we will pay more attention to the extraction strategy of messages. The task of information hiding becomes the challenge of whether the message can be extracted correctly.

Under the framework of cover synthesis, image steganography becomes the task of the space mapping between message space M and stego image space S . The embedding process can be regarded as a message-stego mapping. In contrast, the message extraction can be viewed as a stego-message mapping, as shown in Fig. 22. Because of the randomness of generating stego images based on the generator, the mapping relationship between message and the stego image may be one-to-many, and the goal of steganography is to seek the mapping relationship satisfying the constraints of message loss and fidelity loss.

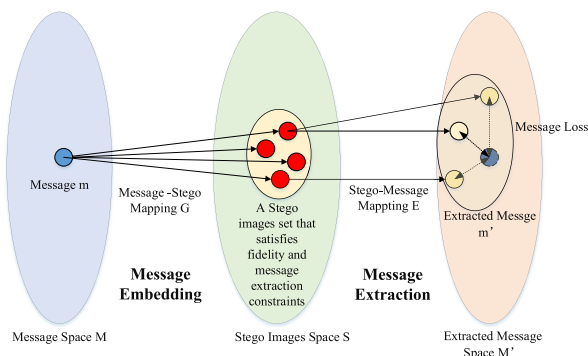


FIGURE 22. A framework for messages embedding and extraction in cover synthesis by GAN.

Similar to the spatial domain and transform domain steganography, the locations of messages in the stego image are different. When the message constraint directly acts on the space domain, cover synthesis can be regarded as spatial domain steganography, such as Liu’s method [96]. The secret message is hidden behind the generated pixels of the image. When the message constraint acts on the transform domain, cover synthesis can be regarded as a transform domain steganography scheme. In Hu’s method [19], they hide the message in noise and need to be recovered by a neural network extractor. Similar to [19], [95] hides the message by the semantic labels. These methods are all steganography

schemes in the transform domain. When the deep neural networks are treated as an encoder, they convert data into a feature space, such as [93]. This method can be regarded as a steganography method in the transform domain. One of the advantages of transform domain steganography is that the encoded messages can contain resistance various image distortions. Although robustness is not the goal of traditional steganography, in some specific situations, it is a practical requirement.

Therefore, in the case of using a neural network or generator, the steganography is converted into an optimization problem of defining a total loss function,

$$L_{total} = \lambda_f L_{fidelity} + \lambda_m L_{message} + \lambda_{ro} L_{robustness} + \lambda_{re} L_{reversible} \quad (25)$$

where $L_{fidelity}$ and $L_{message}$ represent the concerns of traditional steganography: the accurate extraction of the message and the natural properties of the stego image. $L_{robustness}$ and $L_{reversible}$ represent some other properties such as robustness or reversibility in steganography. These Loss weights λ_s indicate the proportion of each performance requirement in different application scenarios.

3) STRATEGY: FROM SUPERVISED TO UNSUPERVISED

According to the various information received by the generator, when generating the stego image, we divide the cover synthesis method into supervised, unsupervised, semi-supervised in Section VI. The semi-supervised manner can be regarded as a general framework for stego image generation. Both supervised and unsupervised modes can be seen as a particular case of a semi-supervised way.

As shown in Fig.23, we can relatively easily grasp the commonalities of these three strategies. In the unsupervised method [19], no cover image can be treated as 100% corruption. Therefore, it is necessary to construct a stego image by utilizing the mapping relationship between message and noise. The message loss constraint is based on noise extraction accuracy. In a semi-supervised method [96], with image completion techniques, secret messages are embedded in uncorrupted image regions, and message loss constraints are built into a portion of the image. In the supervised method [15], [78], due to the existence of the cover image, the constraints of the message loss are based on the difference between the generated stego images and the cover

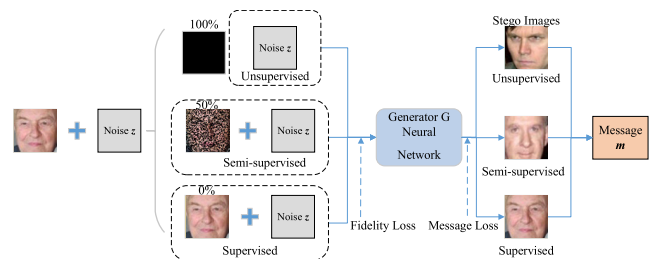


FIGURE 23. A general framework for stego image generation.

images. It should be pointed out that the terms, such as supervised, semi-supervised, and unsupervised, are mainly defined according to the dependence on an explicit cover image, which is not the same as those used in machine learning.

4) TRAINING MODE OF THE GENERATOR

The mechanism of obtaining the synthetic stego image depends on the training mode of the generative model, which can be divided into two implementation strategies. The first strategy is to train a generator with message loss constraint and prior constraint simultaneously, namely parallel constraint synthesis (PCS) mode, as shown in Fig. 24. After the generator is trained, the cover image can be sampled directly from the generator. However, because the message is relevant to the generator. You need to repeatedly prepare a new generator when a new message needs to be hidden. Currently, due to the high cost of training generators, this strategy has significant limitations in practice. To the best of our knowledge, Li’s method [94] follows this framework that utilizes parallel constraint synthesis mode.

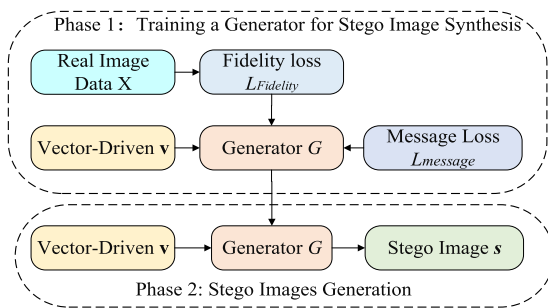


FIGURE 24. Constraint parallel training mode for the cover synthesis.

The second strategy is to satisfy the prior constraint and message constraint through two subsequent schemes, namely sequential constraint synthesis (SCS), as shown in Fig. 25. First, a real image data set is used to train a generator that satisfies the fidelity loss constraint, $L_{Fidelity}(s)$. Then, we can design a generation scheme that meets the message extraction loss constraint, such as $L_{message}(m_{ext}|s, k)$. The character of this scheme is that the limitations of message loss and prior loss can be separated, that is, when training the generator,

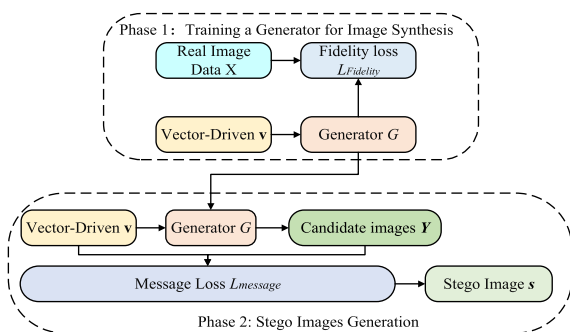


FIGURE 25. Separate serial constraint mode for the cover synthesis.

we only need to pay attention to how to make the generated sample distribution approximate to the real data distribution. After training, the generator is used to construct the candidate stego image, and the final stego image is obtained by using a message constraint. The separability of constraint conditions will make the design of cover synthesis more practical and straightforward. This separation is a specific implementation scheme of the payload-distance sender mode. All of these methods, such as [19], [95], [96] adopt this serial mode.

VII. EVALUATIONS METRICS ON GAN-BASED STEGANOGRAPHY

In this section, we evaluate GAN-based steganography on three axes: *secrecy*, the difficulty of detecting stego images; *capacity*, the number of message bits that can be hidden in the stego image; and *robustness*, the degree to which methods can succeed with some image distortions. All mentioned methods are divided into three types, such as cover modification, selection, and synthesis as before.

A. SECURITY

Steganographic security mainly includes the indistinguishability and computational complexity of obtaining the embedded message. In this section, we start with an image quality evaluation. Then, we compare the statistical indistinguishability of these methods via data-driven steganalysis tools.




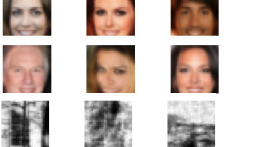



1) IMAGE QUALITY

First of all, it should be noted that the cover selection steganography [16] only selects a cover image as a stego image. We assume that the image quality of this method is perfect. Our discussion is mainly on the practices of constructing stego images by a generator, which includes cover modification and synthesis. We report the experimental results from qualitative and quantitative comparisons.

Qualitative comparisons In Table 1, we present the generators used in the different cover modification methods and the image datasets, as well as the visual effects of the stego images obtained by these methods. As can be seen from the table, those steganographic methods, such as [17], [79], [81], that use the generator to generate the cover image, the resulting stego image quality is not good. This is mainly due to the stego image quality depending on the performance of the generator. And those methods, such as [14], [82], use the GAN to learn a modification probability matrix have higher visual effects because they rely on the cover image.

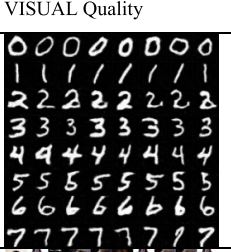

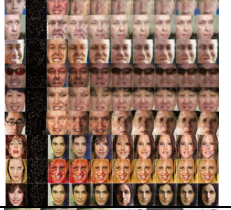
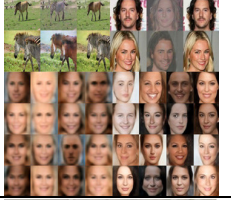

Table 2 shows the synthesis methods for generating a stego image using a generator without relying on a specific cover image. From these methods, it can be seen that the visual quality of the stego image is entirely dependent on the performance of the generator. The generators used by these methods are relatively simple, so the resulting effects are not sound. The one exception is that in CycleGAN-based steganography [99], BEGAN is used to generate images, so it has a higher visual quality.

TABLE 1. Comparisons of images for cover modification with GANs.

Ref.	Generator	Dataset	VISUAL Quality
[79]	GAN	CelebA[101]	
[17]	WGAN	CelebA[101]	
[81]	WGAN	CelebA[101]	
[15]	Encoder	BOSS[102] and CelebA[101]	
[82]	ASDL-GAN	BOSS[102]	
[14]	UT-SCA-GAN	BOSS[102]	
[91]	Critic network	Div2k[103] and COCO[104]	

Quantitative analysis One widely-used metric for measuring the quality of images is the peak signal-to-noise ratio (PSNR) and Structural similarity index (SSIM) [108] between the cover image and the stego image. Since GAN-based cover modification uses LSB-like [109], [110] or minimizes distortion [32] in the modification strategy, it has

TABLE 2. Comparisons of images for cover modification with GANs.

Ref.	GAN model	Dataset	VISUAL Quality
[95]	ACGAN	MNIST[105] and CelebA[101]	
[19]	DCGAN	CelebA[101] and Food101 [106]	
[96]	DCGAN:	LFW[107] and CelebA [101]	
[99]	CycleGAN:	horse2zebra, woman2man [99]	
[98]	CycleGAN	1,000 aerial photographs X and 1,000 maps Y[98]	

been shown that these methods make the PSNR value large, and the image quality difference is small compared with the cover image. The *SteganoGAN* method [91] reports the PSNR and SSIM in their work, where the PSNR values fall between 35-41, and the SSIM values are above 0.9. However, for a method of directly generating a dense image using a generator, there is no one-to-one pixel correspondence. Metrics like PSNR are not suitable to evaluate the stego image. Quantitative indicators for the GAN model often use Fréchet inception distance (FID) [111] and inception score (IS) [5]. Other evaluation criteria include Mode Score [56], Kernel MMD [112], Wasserstein distance, and 1-nearest neighbor (1-NN)-based two data tests [113]. These indicators are still an ongoing important research area.

2) STATISTICAL STEGANALYSIS

Steganographic security is often evaluated using a classifier to distinguish between cover and stego images. In this paper,

we directly adopted the best results reported in their original paper. Since these methods use different ways for steganalysis, we also point out the classifiers they use while giving the detection rate. In this case, although we cannot evaluate the performance of these algorithms objectively, their experimental results will provide us with a relative criterion for the security of these methods.

It can be seen from Table 3, those methods that use the generator to generate the cover image and introduce the steganalyzer have sound security for their steganalysis tools, such as [79], [17]. At the same time, the security of using GAN to construct a modification probability matrix is very close to the traditional steganography by cover modification, such as [82], [14].

TABLE 3. The FIDs of different models trained on CelebA.

Methods	Embedding methods	Classifier	Error rate
SGAN [79]	± 1 embedding	<i>Self-Defined</i>	0.50
SGAN[79]	HUGO	<i>Self-Defined</i>	0.49
SSGAN [17]	± 1 embedding	<i>Self-Defined</i>	0.72
SSGAN [17]	HUGO	Qian’s Net [80]	0.71
ASDL-GAN[82]	minimal-distortion	Xu’s Net [84]	0.27
ASDL-GAN[82]	minimal-distortion	EC+SRM[114, 115]	0.26
UT-SCA-GAN[14]	minimal-distortion	EC+SRM[114, 115]	0.22.
UT-SCA-GAN[14]	minimal-distortion	EC+maxSRMd2[114, 115]	0.21
Tang[88]	Adversarial Embedding	Xu’s Net [84]	0.58

In Table 4, we list the error rate of statistical analysis of steganography by cover synthesis with GAN. Based on the cover image as an input, those methods that use the adversarial game strategy to generate stego images, such as [15], [91], also have a certain degree of security. SWE’s [19] case 1 assumes that the attacker is unable to obtain training samples, and the security is higher at this time.

TABLE 4. The FIDs of different models trained on CelebA.

Methods	Embedding methods	Classifier	Error rate
Hayes[15]	Adversarial Training	<i>Self-Defined</i>	0.79
Zhang[91]	Adversarial Training	Ye et al.[116]	0.50
SWE[19] case 1	Message-noise	Ni’s model [116]	0.53
SWE[19] case 2	Message-noise	Ni’s model[116]	0.02
ACGAN[95]	Message-label	EC+SPAM[114, 117]	0.52
GSS[96]	Message Loss	EC+SPAM[114, 117]	0.42
GSS[96]	Message Loss	EC+SCRMQ1[114, 118]	0.04
CycleGAN [99]	Message-noise	Ni’s model [116]	0.54

Still, in case 2, when directly using training images to train the steganalyzer, steganalysis achieves good detection ability. The problem with this approach is that the steganographic analysis becomes forensic of the composite image at this time, that is, whether the image is synthetic. ACGAN-based method [95] considers cases where training samples cannot be obtained. In GSS [96] method, under the embedded capacity of 0.4bpp, the security is higher for SPAM features, but for SCRMQ1 [118], the classifier gets a good detection ability. The benefit of this sampling method is that the training set can be exposed, and the steganographic security can depend on the confidentiality of the embedded key.

3) SECURITY LEVELS WITH KERCKHOFFS’S PRINCIPLE

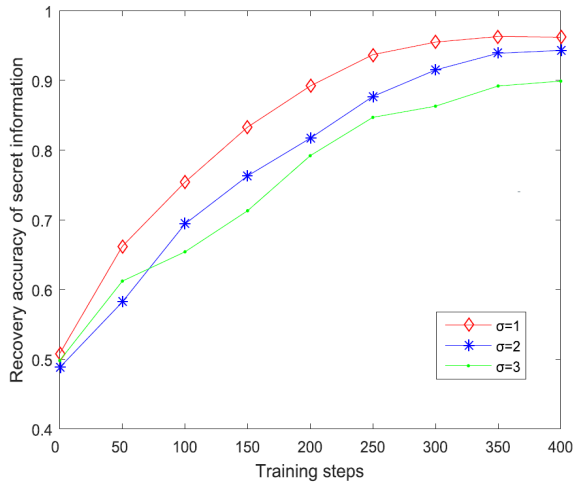
Ke *et al.* [28] proposed a stego-security classification strategy with Kerckhoffs’s principle based on the different levels of steganalysis attacks such as Stego-Cover Only Attack (SCOA), Known Cover Attack (KCA), Chosen Cover Attack (CCA) and Adaptive Chosen Cover Attack. In synthesis methods, such as [19], [95], there are explicitly extraction or embedding key *k*. The mapping itself can be used as a key, but in this case, the keyspace is too small to resist SOA attacks. Therefore, when the algorithm exposes an active attack environment that directly attempts to extract a key, it is not secure in terms of the computational complexity of acquiring keys. In the GSS method [96], the keyspace meets the specified computational complexity when the size of the Cardan grille is large enough. Therefore, the GSS method can be stego-secure against SCOA. The training image set should be available for the attacker in the KCA model. In [19], it has been shown that directly using the training set to train classifiers for steganalysis is unsafe. Therefore, the cover synthesis method is not stego-secure against KCA. At present, the actual security requirements for cover synthesis are as follows. 1) the training dataset and the key *k* should be kept secrecy. 2) $|\mathcal{K}|$ should be large enough to meet requirements of computational complexity.

B. CAPACITY AND RECOVERY ACCURACY

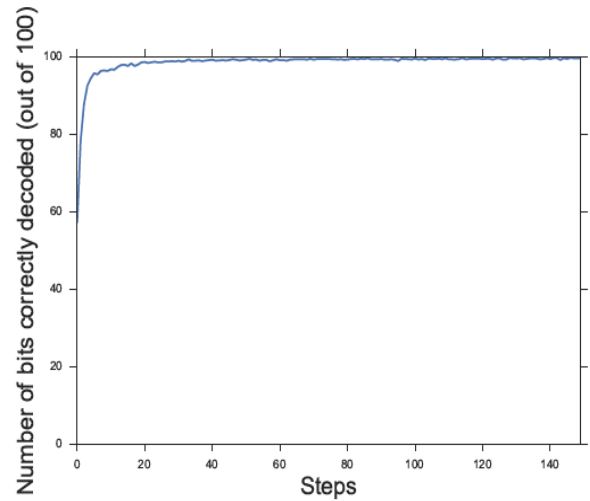
At present, there is still a big gap in performance between GAN-based steganography by cover synthesis and traditional cover modification methods. It has reached a considerable level compared to the traditional cover selection or synthesis method. We list the capacity of cover synthesis methods by GAN in Table 5, and the absolute capacity is shown in the second column, the size of the stego image is listed in the third column, the relative capacity is shown in the last column:

$$Relative\ capacity = \frac{Number\ of\ message\ bits}{Size\ of\ the\ image} \quad (26)$$

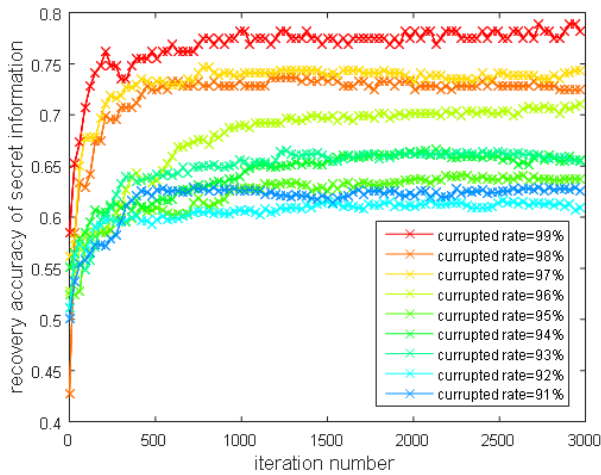
Extraction rate. However, in those schemes, such as [15], [19], [94]–[96], where the generator directly generates the stego image, the stego image generation depends on the optimization problem of the neural network, that is, the minimization of a certain cost function. Since the



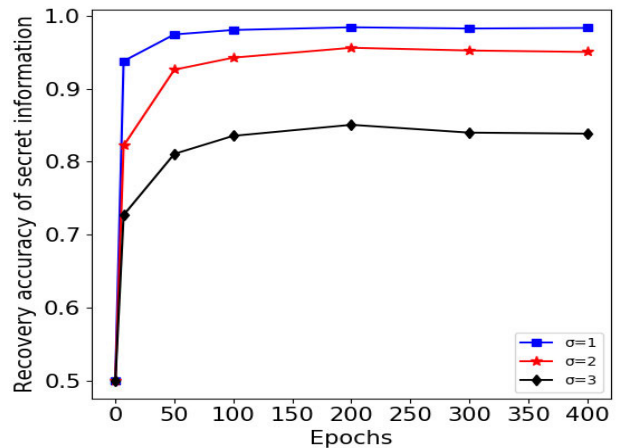
(a) SWE



(b) ACGAN-Stego



(c) GSS



(d) the recovery accuracy of [94]

FIGURE 26. Extraction rate for the cover synthesis.

TABLE 5. Capacities of various non-modification methods.

Reference	Absolute Capacity (bytes/image)	Image Size	Relative Capacity (bytes/pixel)
[119]	3.72	$\geq 512 \times 512$	$1.42e-5$
[43]	1.125	512×512	$4.29e-6$
[120]	2.25	512×512	$8.58e-6$
[51]	64×64	800×800	$6.40e-3$
[49]	1535~4300	1024×1024	$1.46e-3 \sim 4.10e-3$
[19]	≥ 37.5	64×64	$9.16e-3$
[95]	0.375	32×32	$3.7e-4$
[96]	18.3-135.4	64×64	$1.49e-3 \sim 1.10e-2$

generator or neural network usually cannot get the optimal solution, the message may not be extracted correctly. The actual capacity can be denoted as:

$$Actual\ capacity = Relative\ capacity \times Extraction\ rate \tag{27}$$

In Fig.26, we show the extraction rate of different algorithms when the message cannot be extracted exactly.

It can be seen from the Fig.26 that in the methods of (a) [19] and (b) [95], additional training is required, and the message extractor gradually increases the stability of the message extraction as the number of training steps increases. The ACGAN-based method message is hidden in the label of the image, that is, the semantic level. By training a classifier, the category information can be obtained, so that the message extraction accuracy is high, and the disadvantage is that the sneaking rate is low. In Fig.26 (c) [96], to verify the accuracy of message extraction, we first perform random damage, according to the damage ratio of 99%-91%, and embed the message on all pixels that are not damaged. The recovery accuracy increases as the iteration numbers increases. Due to limitations in learning performance that the message constraint cannot be completely satisfied. The actual embedding capacity is not high. In Fig.26 (d), the recovery accuracy [94] increases as the number of training steps increases. After about ten epochs (every epoch has 633 steps), the recovery accuracy rapidly increased to a higher level.

C. ROBUSTNESS

In the steganography by cover synthesis, such as [19] and [95], [121], the message is hidden in the transform domain of the generated image, so that it has certain robustness. Furthermore, varying the types of image distortion during the training process, [92] and [93] show that steganography model can learn robustness to a variety of different image distortions. In this section, we only test the robustness of [19] and [121] with common image attacks. We consider applying four typical image attacks. These attack conditions are listed as follows.

C1. Contrast enhancement by multiplying the intensity of the image pixels with factors of 1.1 and 1.5

C2. Gaussian noise addition (variance 0.01).

C3. Salt noise added (density 0.05).

C4. JPEG compression with varying quality facts (q.f. 90, q.f. 60 and q.f. 30).

We use the G network to generate 5,000 stego images based on the CIFA-100 dataset and apply the four typical methods to attack each group of images. Then, we give the accuracy of the message extraction after the attack, for Hu *et al.* [19], the result is shown when parameter σ is 3, and δ is 0.001. A group of results is shown in Table 6.

TABLE 6. Extraction accuracy of an extractor for the attacked stego images.

Attack	C1		C2	C3	C4		
	1.1 times	1.5 times			q.f. 50	q.f. 70	q.f. 90
[121] accuracy (%)	100	98	98.9 9	99.9 6	98.52	99.89	100
[19] accuracy (%)	81.72	71.23	54.4 9	53.6 8	61.33	62.82	66.10

From the experimental results, these methods are robust to all four attacks, especially the method of Zhang *et al.* [121], able to resist jpeg compression and contrast enhancement. The model has no errors at a jpeg compression factor of 90, and brightness changes at an intensity of 1.1 times because the message relies on the recognition of image semantic labels by neural networks. The neural networks have good fault tolerance. When inputting fuzzy or incomplete information, a suboptimal approximate solution can be given to achieve the correct identification of incomplete input information. The noise extracted from the image has no clear semantic meaning. However, it can be seen from the experimental results that the noise can be resistant to contrast enhancement, but is less robust to JPEG compression. This is consistent with our idea of treating this method as a kind of information hiding in the transform domain.

VIII. PROSPECTIVE AND CONCLUSION

A. PERSPECTIVE

This paper reviews recent researches on image steganography based on GAN. At present, the cover modified method has distinct advantages in terms of embedded capacity, anti-statistical analysis, and message capacity. The performance of GAN-based steganography is far from that of traditional methods in many aspects. We believe that the methods based on GAN will be a promising field of research in steganography. At present, for the GAN-based steganography method, the following aspects need to be further improved.:

1) STEGANOGRAPHY CAPACITY

In GAN-CSY, such as [96], the message extraction is performed directly on the image pixel in the spatial domain. The instability of generated pixels leads to the low accuracy of message extraction. The message in method [19], [95] does not exist on the pixel value itself but exists as a category attribute [95] or a noise vector [19]. The disadvantage is that the embedded capacity is low. Improvement of message stability or embedding capacity will be the focus of future research.

2) IMAGE EVALUATION

It is tough to quantify the quality of synthetic images, in the field of image synthesis, the evaluation criteria of the generated images are not sound enough. Some methods using manual evaluation are subjective and lack of objective evaluation criteria. The current evaluation criteria are mainly IS (Inception score) and Frechet Inception Distance (FID). These methods only consider the authenticity and quality of the image. These indicators are still an ongoing important research area.

3) STEGANALYSIS

Under the framework of GAN-CSY, the task of steganalysis is divided into two phases. The first phase is the image forensics, which will tell us whether or not the image is fake. The second stage is image steganalysis, which detects whether or not the generated image contains a secret message. Currently, the images generated by GAN are indistinguishable for human vision. Many image forensics methods can be used to distinguish between natural cover images and generated stego images. In the future, using a computer to synthesize images, videos, or other media will be common. Hiding messages in the generated images will become a new type of covert communication means. In this case, it can be difficult to tell whether the generated image is stego or not. Improving the performance of steganalysis will be one of the potential areas for future research.

B. CONCLUSION

With the generative models, image steganography began to merge with the field of computer vision. Traditional computer

vision researchers have also started to study image steganography [92]. A combination of the research fields broadens the areas of image steganography. Besides, the introduction of GANs into the research ideas of information hiding will also have a significant impact on the development of other information hiding technologies, such as digital watermarking technology.

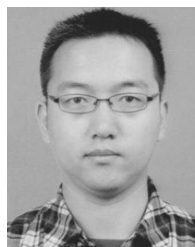
In this paper, we review image steganography with GAN. Firstly, we give the principle and characteristics of steganography. Then the traditional image steganography method and the problems are discussed. We also introduce the principle and some improvement models of GAN. This paper focuses on the GAN-based steganography with cover modification, cover selection, and cover synthesis. We analyzed the different roles of GAN in these methods. The GAN-based cover modification methods use GAN to construct the cover image or a modification matrix. The GAN-based cover selection method has a low embedding capacity and requires a secret channel to pass the key. The GAN-based cover synthesis methods directly use the generator trained by GAN to obtain the stego images. The GAN-based cover synthesis methods are divided into three categories, unsupervised, semi-supervised, and supervised methods for discussion. We also give evaluation criteria of GAN-based steganography with secrecy, capacity, and robustness. In conclusion, for the long-term development of image steganography, using GAN to enhance the abilities of steganographer to design a more safety and efficiency methods is a question worth studying.

REFERENCES

- [1] J. Fridrich, *Steganography in Digital Media: Principles, Algorithms, and Applications*, 1st ed. New York, NY, USA: Cambridge Univ. Press, 2009.
- [2] X. Lu, Y. Wang, L. Huang, W. Yang, and Y. Shen, "A secure and robust covert channel based on secret sharing scheme," in *Proc. APWeb, Web Technol. Appl.* Cham, Switzerland, 2016, pp. 276–288.
- [3] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," 2014, *arXiv:1406.2661*. [Online]. Available: <https://arxiv.org/abs/1406.2661>
- [4] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2016, *arXiv:1511.06434*. [Online]. Available: <http://arxiv.org/abs/1511.06434>
- [5] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," 2016, *arXiv:1606.03498*. [Online]. Available: <https://arxiv.org/abs/1606.03498>
- [6] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," 2017, *arXiv:1701.07875*. [Online]. Available: <https://arxiv.org/abs/1701.07875>
- [7] D. Berthelot, T. Schumm, and L. Metz, "BEGAN: Boundary equilibrium generative adversarial networks," 2017, *arXiv:1703.10717*. [Online]. Available: <http://arxiv.org/abs/1703.10717>
- [8] X. Chen, Y. Duan, R. Houthoof, J. Schulman, I. Sutskever, and P. Abbeel, "InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets," 2016, *arXiv:1606.03657*. [Online]. Available: <http://arxiv.org/abs/1606.03657>
- [9] J. Zhao, M. Mathieu, and Y. Lecun, "Energy-based generative adversarial network," 2016, *arXiv:1609.03126*. [Online]. Available: <https://arxiv.org/abs/1609.03126>
- [10] L. Yu, W. Zhang, J. Wang, and Y. Yu, "SeqGAN: Sequence generative adversarial nets with policy gradient," 2017, *arXiv:1609.05473*. [Online]. Available: <http://arxiv.org/abs/1609.05473>
- [11] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: An overview," 2017, *arXiv:1710.07035*. [Online]. Available: <http://arxiv.org/abs/1710.07035>
- [12] Y.-J. Cao, L.-L. Jia, Y.-X. Chen, N. Lin, C. Yang, B. Zhang, Z. Liu, X.-X. Li, and H.-H. Dai, "Recent advances of generative adversarial networks in computer vision," *IEEE Access*, vol. 7, pp. 14985–15006, 2019.
- [13] H. Huang, P. S. Yu, and C. Wang, "An introduction to image synthesis with generative adversarial nets," 2018, *arXiv:1803.04469*. [Online]. Available: <http://arxiv.org/abs/1803.04469>
- [14] J. Yang, K. Liu, X. Kang, E. K. Wong, and Y.-Q. Shi, "Spatial image steganography based on generative adversarial network," 2018, *arXiv:1804.07939*. [Online]. Available: <http://arxiv.org/abs/1804.07939>
- [15] J. Hayes and G. Danezis, "Generating steganographic images via adversarial training," 2017, *arXiv:1703.00371*. [Online]. Available: <http://arxiv.org/abs/1703.00371>
- [16] Y. Ke, M. Zhang, J. Liu, T. Su, and X. Yang, "Generative steganography with Kerckhoffs' principle," *Multimedia Tools Appl.*, vol. 78, no. 10, pp. 13805–13818, 2019.
- [17] H. Shi, J. Dong, W. Wang, Y. Qian, and X. Zhang, "SSGAN: Secure steganography based on generative adversarial networks," 2018, *arXiv:1707.01613*. [Online]. Available: <http://arxiv.org/abs/1707.01613>
- [18] J. Liu, T. Zhou, Z. Zhang, Y. Ke, Y. Lei, M. Zhang, and X. Yang, "Digital cardan grille: A modern approach for information hiding," 2018, *arXiv:1803.09219*. [Online]. Available: <http://arxiv.org/abs/1803.09219>
- [19] D. Hu, L. Wang, W. Jiang, S. Zheng, and B. Li, "A novel image steganography method via deep convolutional generative adversarial networks," *IEEE Access*, vol. 6, pp. 38303–38314, 2018.
- [20] S. Baluja, "Hiding images in plain sight: Deep steganography," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 2069–2079.
- [21] W. Tang, B. Li, S. Tan, M. Barni, and J. Huang, "CNN-based adversarial embedding for image steganography," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 8, pp. 2074–2087, Aug. 2019.
- [22] G. J. Simmons, "The prisoners' problem and the subliminal channel," in *Proc. Crypto*. Boston, MA, USA: Springer, 1983.
- [23] A. L. von and N. J. Hopper, "Public-key steganography," in *Proc. EURO-CRYP*. Berlin, Germany: Springer, 2004, pp. 323–341.
- [24] B. Li, J. He, J. Huang, and Y. Q. Shi, "A survey on image steganography and steganalysis," *J. Inf. Hiding Multimedia Signal Process.*, vol. 2, no. 2, pp. 288–289, 2011.
- [25] C. E. Shannon, "Communication theory of secrecy systems," *Bell Syst. Tech. J.*, vol. 28, no. 4, pp. 656–715, 1948.
- [26] N. J. Hopper, J. Langford, and L. Von Ahn, "Provably secure steganography," in *Proc. 22nd Annu. Int. Cryptol. Conf. Adv. Cryptol.* Berlin, Germany: Springer, 2002, pp. 77–92.
- [27] S. Katzenbeisser and F. A. P. Petitcolas, "Defining security in steganographic systems," *Proc. SPIE*, vol. 4675, pp. 50–56, Apr. 2002.
- [28] Y. Ke, J. Liu, M.-Q. Zhang, T.-T. Su, and X.-Y. Yang, "Steganography security: Principle and practice," *IEEE Access*, vol. 6, pp. 73009–73022, 2018.
- [29] C. Cachin, "An information-theoretic model for steganography," *Inf. Comput.*, vol. 192, no. 1, pp. 41–56, Jul. 2014.
- [30] R. Chandramouli, M. Kharrazi, and N. Memon, "Image steganography and steganalysis: Concepts and practice," in *Digital Watermarking (Lecture Notes in Computer Science)*, vol. 2939. Berlin, Germany: Springer, 2004, pp. 35–49.
- [31] T. Pevný and J. Fridrich, "Benchmarking for Steganography," in *Proc. Inf. Hiding, Int. Workshop*, Santa Barbara, CA, USA, 2008, pp. 251–267.
- [32] T. Filler, J. Judas, and J. Fridrich, "Minimizing additive distortion in steganography using syndrome-trellis codes," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 3, pp. 920–935, Sep. 2011.
- [33] A. J. L. Westfeld, "F5—A steganographic algorithm: High capacity despite better steganalysis," in *Information Hiding (Lecture Notes in Computer Science)*, vol. 2137, I. S. Moskowitz, Eds. Berlin, Germany: Springer, 2001, pp. 289–302.
- [34] K. Solanki and A. B. S. Sarkar Manjunath, "YASS: Yet another steganographic scheme that resists blind steganalysis," in *Proc. Inf. Hiding, Int. Workshop (IH)*, Saint-Malo, France, Jun. 2007, pp. 16–31.
- [35] V. Holub and J. Fridrich, "Digital image steganography using universal distortion," in *Proc. 1st ACM Workshop Inf. Hiding Multimedia Secur. (IH&MMSec)*, Montpellier, France, 2013, pp. 59–68.
- [36] B. Diouf, I. Diop, S. M. Farssi, and O. Khouma, "Minimizing embedding impact in steganography using polar codes," in *Proc. Int. Conf. Multimedia Comput. Syst. (ICMCS)*, Marrakech, Morocco, Apr. 2014, pp. 105–111.

- [37] V. Holub, J. Fridrich, and T. Denemark, "Universal distortion function for steganography in an arbitrary domain," *EURASIP J. Inf. Secur.*, vol. 2014, no. 1, p. 1, Dec. 2014.
- [38] B. Li, M. Wang, J. Huang, and X. Li, "A new cost function for spatial image steganography," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Paris, France, Oct. 2014, pp. 4206–4210.
- [39] V. U. Sameer and R. Naskar, "Universal wavelet relative distortion: A new counter-forensic attack on photo response non-uniformity based source camera identification," in *Proc. Inf. Secur. Pract. Exper. (ISPEC)*, Cham, Switzerland: Springer, 2018, pp. 37–49.
- [40] H. Sajedi and M. Jamzad, "Secure cover selection steganography," in *Advances in Information Security and Assurance*. Berlin, Germany: Springer, 2009, pp. 317–326.
- [41] R. E. Yang, Z. W. Zheng, and W. Jin, "Cover selection for image steganography based on image characteristics," *J. Optoelectron. Laser*, vol. 25, no. 4, pp. 764–768, 2014.
- [42] Y. Sun and F. Liu, "Selecting cover for image steganography by correlation coefficient," in *Proc. 2nd Int. Workshop Edu. Technol. Comput. Sci.*, Wuhan, China, 2010, pp. 159–162.
- [43] Z. Zhou, H. Sun, R. Harit, X. Chen, and X. Sun, "Coverless image steganography without embedding," in *Cloud Computing and Security*. Cham, Switzerland: Springer, 2015, pp. 123–132.
- [44] F.-F. Li and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, San Diego, CA, USA, vol. 2, Jun. 2005, pp. 524–531.
- [45] L.-Y. Wei, "Deterministic texture analysis and synthesis using tree structure vector quantization," in *Proc. 12th Brazilian Symp. Comput. Graph. Image Process.*, Campinas, Brazil, 1999, pp. 207–213.
- [46] Q. Zhao, A. K. Jain, N. G. Paulter, and M. Taylor, "Fingerprint image synthesis based on statistical feature models," in *Proc. IEEE 5th Int. Conf. Biometrics, Theory, Appl. Syst. (BTAS)*, Arlington, VA, USA, Sep. 2012, pp. 23–30.
- [47] H. Otori and S. Kuriyama, "Data-embeddable texture synthesis," in *Smart Graphics*. Berlin, Germany: Springer, 2007, pp. 146–157.
- [48] H. Otori and S. Kuriyama, "Texture synthesis for mobile data communications," *IEEE Comput. Graph. Appl.*, vol. 29, no. 6, pp. 74–81, Nov./Dec. 2009.
- [49] K. C. Wu and C. M. Wang, "Steganography using reversible texture synthesis," *Int. J. Eng. Technol.*, vol. 7, no. 1, pp. 130–139, Jan. 2015.
- [50] Z. Qian, H. Zhou, W. Zhang, and X. Zhang, "Robust steganography using texture synthesis," in *Proc. 12th Int. Conf. Intell. Inf. Hiding Multimedia Signal Process.*, 2016, pp. 25–33.
- [51] J. Xu, X. Mao, X. Jin, A. Jaffer, S. Lu, L. Li, and M. Toyoura, "Hidden message in a deformation-based texture," *Vis. Comput.*, vol. 31, no. 12, pp. 1653–1669, Dec. 2015.
- [52] L. Pan, Z. X. Qian, and X. P. Zhang, "Steganography by constructing texture images," *J. Appl. Sci.*, vol. 34, no. 5, pp. 625–632, 2016.
- [53] S. Li and X. Zhang, "Toward construction-based data hiding: From secrets to fingerprint images," *IEEE Trans. Image Process.*, vol. 28, no. 3, pp. 1482–1497, Mar. 2019.
- [54] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*. [Online]. Available: <http://arxiv.org/abs/1411.1784>
- [55] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier GANs," 2016, *arXiv:1610.09585*. [Online]. Available: <http://arxiv.org/abs/1610.09585>
- [56] T. Che, Y. Li, A. Paul Jacob, Y. Bengio, and W. Li, "Mode regularized generative adversarial networks," 2016, *arXiv:1612.02136*. [Online]. Available: <http://arxiv.org/abs/1612.02136>
- [57] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, "Improved training of wasserstein GANs," 2017, *arXiv:1704.00028*. [Online]. Available: <http://arxiv.org/abs/1704.00028>
- [58] R. Yeh, C. Chen, and T. Y. Lim, M. Hasegawa-Johnson, and M. N. Do, "Semantic image inpainting with perceptual and contextual losses," 2016, *arXiv:1607.07539v2*. [Online]. Available: <https://arxiv.org/abs/1607.07539v2>
- [59] M. Yang, W. Zhao, W. Xu, Y. Feng, Z. Zhao, X. Chen, and K. Lei, "Multitask learning for cross-domain image captioning," *IEEE Trans. Multimedia*, vol. 21, no. 4, pp. 1047–1061, Apr. 2019.
- [60] X. Liang, Z. Hu, H. Zhang, C. Gan, and E. P. Xing, "Recurrent topic-transition GAN for visual paragraph generation," 2017, *arXiv:1703.07022*. [Online]. Available: <http://arxiv.org/abs/1703.07022>
- [61] X. Wang, A. Shrivastava, and A. Gupta, "A-Fast-RCNN: Hard positive generation via adversary for object detection," 2017, *arXiv:1704.03414*. [Online]. Available: <http://arxiv.org/abs/1704.03414>
- [62] P. Luc, C. Couprie, S. Chintala, and J. Verbeek, "Semantic segmentation using adversarial networks," 2016, *arXiv:1611.08408*. [Online]. Available: <http://arxiv.org/abs/1611.08408>
- [63] S. Rajeswar, S. Subramanian, F. Dutil, C. Pal, and A. Courville, "Adversarial generation of natural language," 2017, *arXiv:1705.10929*. [Online]. Available: <http://arxiv.org/abs/1705.10929>
- [64] J. Li, W. Monroe, T. Shi, S. Jean, A. Ritter, and D. Jurafsky, "Adversarial learning for neural dialogue generation," 2017, *arXiv:1701.06547*. [Online]. Available: <http://arxiv.org/abs/1701.06547>
- [65] A. Cherian and A. Sullivan, "Sem-GAN: Semantically-consistent image-to-image translation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Waikoloa Village, HI, USA, Jan. 2019, pp. 1797–1806.
- [66] S. Nowozin, B. Cseke, and R. Tomioka, "F-GAN: Training generative neural samplers using variational divergence minimization," 2016, *arXiv:1606.00709*. [Online]. Available: <http://arxiv.org/abs/1606.00709>
- [67] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text to image synthesis," 2016, *arXiv:1605.05396*. [Online]. Available: <http://arxiv.org/abs/1605.05396>
- [68] S. Reed, Z. Akata, S. Mohan, S. Tenka, B. Schiele, and H. Lee, "Learning what and where to draw," 2016, *arXiv:1610.02454*. [Online]. Available: <http://arxiv.org/abs/1610.02454>
- [69] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. Metaxas, "StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks," 2016, *arXiv:1612.03242*. [Online]. Available: <http://arxiv.org/abs/1612.03242>
- [70] A. Nguyen, J. Clune, Y. Bengio, A. Dosovitskiy, and J. Yosinski, "Plug & play generative networks: Conditional iterative generation of images in latent space," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jun. 2017, pp. 3510–3520.
- [71] Y. Jing, Y. Yang, Z. Feng, J. Ye, Y. Yu, and M. Song, "Neural style transfer: A review," 2017, *arXiv:1705.04058*. [Online]. Available: <http://arxiv.org/abs/1705.04058>
- [72] X. Liang, L. Lee, W. Dai, and E. P. Xing, "Dual motion GAN for future-flow embedded video prediction," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1744–1752.
- [73] J. Li, X. Yang, X. Liao, F. Pan, and M. Zhang, "A game-theoretic method for designing distortion function in spatial steganography," *Multimedia Tools Appl.*, vol. 76, no. 10, pp. 12417–12431, May 2016.
- [74] J. M. Ettinger, "Steganalysis and game equilibria," in *Information Hiding*. Berlin, Germany: Springer, 1998, pp. 319–328.
- [75] A. D. Ker, "Batch steganography and the threshold game," *Proc. SPIE*, vol. 6505, Mar. 2007, Art. no. 650504.
- [76] T. Filler and J. Fridrich, "Design of adaptive steganographic schemes for digital images," *Electron. Imag.*, vol. 7880, pp. 181–197, Feb. 2011.
- [77] S. Kouider, M. Chaumont, and W. Puech, "Technical points about adaptive steganography by oracle (ASO)," in *Proc. 20th Eur. Signal Process. Conf. (EUSIPCO)*, Bucharest, Romania, 2012, pp. 1703–1707.
- [78] Y. Zhang, W. Zhang, K. Chen, J. Liu, Y. Liu, and N. Yu, "Adversarial examples against deep neural network based steganalysis," in *Proc. 6th ACM Workshop Inf. Hiding Multimedia Secur. (IH&MMSec)*, 2018, pp. 67–72.
- [79] D. Volkhonskiy, I. Nazarov, and E. Burnaev, "Steganographic generative adversarial networks," 2017, *arXiv:1703.05502*. [Online]. Available: <http://arxiv.org/abs/1703.05502>
- [80] Y. Qian, J. Dong, W. Wang, and T. Tan, "Deep learning for steganalysis via convolutional neural networks," in *Proc. Media Watermarking, Secur., Forensics*, Mar. 2015, Art. no. 94090J.
- [81] Y. Wang, K. Niu, and X. Yang, "Information hiding scheme based on generative adversarial network," *J. Comput. Appl.*, vol. 38, no. 10, pp. 2923–2928, 2018.
- [82] W. Tang, S. Tan, B. Li, and J. Huang, "Automatic steganographic distortion learning using a generative adversarial network," *IEEE Signal Process. Lett.*, vol. 24, no. 10, pp. 1547–1551, Oct. 2017.
- [83] J. Fridrich and T. Filler, "Practical methods for minimizing embedding impact in steganography," *Proc. SPIE*, vol. 6505, Feb. 2007, Art. no. 650502.
- [84] G. Xu, H.-Z. Wu, and Y.-Q. Shi, "Structural design of convolutional neural networks for steganalysis," *IEEE Signal Process. Lett.*, vol. 23, no. 5, pp. 708–712, May 2016.

- [85] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2015, pp. 234–241.
- [86] M. Chaumont, "Deep learning in steganography and steganalysis from 2015 to 2018," 2019, *arXiv:1904.01444*. [Online]. Available: <http://arxiv.org/abs/1904.01444>
- [87] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," 2014, *arXiv:1412.6572*. [Online]. Available: <http://arxiv.org/abs/1412.6572>
- [88] W. Tang, B. Li, S. Tan, M. Barni, and J. Huang, "CNN based adversarial embedding with minimum alteration for image steganography," 2018, *arXiv:1803.09043*. [Online]. Available: <http://arxiv.org/abs/1803.09043>
- [89] S. Ma, Q. Guan, X. Zhao, and Y. Liu, "Adaptive spatial steganography based on probability-controlled adversarial examples," 2018, *arXiv:1804.02691*. [Online]. Available: <http://arxiv.org/abs/1804.02691>
- [90] M. Abadi and D. G. Andersen, "Learning to protect communications with adversarial neural cryptography," 2016, *arXiv:1610.06918*. [Online]. Available: <http://arxiv.org/abs/1610.06918>
- [91] K. Alex Zhang, A. Cuesta-Infante, L. Xu, and K. Veeramachaneni, "SteganoGAN: High capacity image steganography with GANs," 2019, *arXiv:1901.03892*. [Online]. Available: <http://arxiv.org/abs/1901.03892>
- [92] J. Zhu, R. Kaplan, J. Johnson, and L. Fei-Fei, "HiDDeN: Hiding data with deep networks," 2018, *arXiv:1807.09937*. [Online]. Available: <http://arxiv.org/abs/1807.09937>
- [93] M. Tancik, B. Mildenhall, and R. Ng, "StegaStamp: Invisible hyperlinks in physical photographs," 2019, *arXiv:1904.05343*. [Online]. Available: <https://arxiv.org/abs/1904.05343>
- [94] J. Li, J. Liu, G. Su, M. Zhang, and Y. Yang, "An generative steganography method based on WGAN-GP," in *Proc. 2nd Int. Conf. Artif. Intell. Secur.*, Hohhot, China, 2020.
- [95] M.-m. Liu, M.-q. Zhang, J. Liu, Y.-n. Zhang, and Y. Ke, "Coverless information hiding based on generative adversarial networks," 2017, *arXiv:1712.06951*. [Online]. Available: <http://arxiv.org/abs/1712.06951>
- [96] Z. Zhang, J. Liu, Y. Ke, Y. Lei, J. Li, M. Zhang, and X. Yang, "Generative steganography by sampling," *IEEE Access*, vol. 7, pp. 118586–118597, 2019.
- [97] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," 2017, *arXiv:1703.10593*. [Online]. Available: <http://arxiv.org/abs/1703.10593>
- [98] C. Chu, A. Zhmoginov, and M. Sandler, "CycleGAN, a master of steganography," 2017, *arXiv:1712.02950*. [Online]. Available: <http://arxiv.org/abs/1712.02950>
- [99] Z. Zhang, G. Fu, F. Di, C. Li, and J. Liu, "Generative reversible data hiding by image to image translation via GANs," 2019, *arXiv:1905.02872*. [Online]. Available: <https://arxiv.org/abs/1905.02872>
- [100] A. S. Babu, NRIIT, P. N. B. Swamy, and P. C. Rao, "Reversible data hiding in encrypted images by reversible image transformation," *Int. J. Eng. Trends Technol.*, vol. 64, no. 1, pp. 24–30, 2016.
- [101] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," 2014, *arXiv:1411.7766*. [Online]. Available: <http://arxiv.org/abs/1411.7766>
- [102] P. Bas, T. Filler, and T. Pevný, "'Break our steganographic system': The ins and outs of organizing BOSS," in *Information Hiding*. Berlin, Germany: Springer, 2011, pp. 59–70.
- [103] A. Almohammad and G. Ghinea, "Stego image quality and the reliability of PSNR," in *Proc. 2nd Int. Conf. Image Process. Theory, Tools Appl.*, Paris, France, Jul. 2010, pp. 215–220.
- [104] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. Lawrence Zitnick, "Microsoft COCO: Common objects in context," in *Proc. ECCV*, 2014, pp. 740–755.
- [105] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
- [106] L. Bossard, M. Guillaumin, and L. Van Gool, "Food-101—mining discriminative components with random forests," in *Proc. ECCV*, 2014, pp. 446–461.
- [107] E. Learned-Miller, G. B. Huang, A. RoyChowdhury, H. Li, and G. Hua, "Labeled faces in the wild: A survey," in *Advances in Face Detection and Facial Image Analysis*. Cham, Switzerland: Springer, 2016, pp. 189–248.
- [108] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
- [109] J. Mielikainen, "LSB matching revisited," *IEEE Signal Process. Lett.*, vol. 13, no. 5, pp. 285–287, May 2006.
- [110] W. Luo, F. Huang, and J. Huang, "Edge adaptive image steganography based on LSB matching revisited," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 2, pp. 201–214, Jun. 2010.
- [111] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, G. Klambauer, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a Nash equilibrium," 2018, *arXiv:1706.08500v2*. [Online]. Available: <https://arxiv.org/abs/1706.08500v2>
- [112] A. Gretton, K. Borgwardt, M. J. Rasch, B. Scholkopf, and A. J. Smola, "A kernel method for the two-sample problem," 2008, *arXiv:0805.2368*. [Online]. Available: <https://arxiv.org/abs/0805.2368>
- [113] D. Lopez-Paz and M. Oquab, "Revisiting classifier two-sample tests," 2016, *arXiv:1610.06545*. [Online]. Available: <http://arxiv.org/abs/1610.06545>
- [114] J. Kodovsky, J. Fridrich, and V. Holub, "Ensemble classifiers for steganalysis of digital media," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 2, pp. 432–444, Apr. 2012.
- [115] J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 868–882, Jun. 2012.
- [116] J. Ye, J. Ni, and Y. Yi, "Deep learning hierarchical representations for image steganalysis," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 11, pp. 2545–2557, Nov. 2017.
- [117] T. Pevny, P. Bas, and J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 2, pp. 215–224, Jun. 2010.
- [118] M. Goljan, J. Fridrich, and R. Cogranne, "Rich model for steganalysis of color images," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Atlanta, GA, USA, Dec. 2014, pp. 185–190.
- [119] Z. L. Zhou, Y. Cao, and X. M. Sun, "Coverless information hiding based on bag-of-words model of image," *J. Appl. Sci.*, vol. 34, no. 5, pp. 527–536, 2016.
- [120] S. Zheng, L. Wang, B. Ling, and D. Hu, "Coverless information hiding based on robust image hashing," in *Intelligent Computing Methodologies*. Cham, Switzerland: Springer, 2017, pp. 536–547.
- [121] Z. Zhang, G. Fu, J. Liu, and W. Fu, "Generative information hiding method based on adversarial networks," in *Proc. 8th Int. Conf. Comput. Eng. Netw.*, 2020, pp. 261–270.
- [122] X. Hou, L. Shen, K. Sun, and G. Qiu, "Deep feature consistent variational autoencoder," 2017, *arXiv:1610.00291*. [Online]. Available: <https://arxiv.org/abs/1610.00291>
- [123] D. P. Kingma and P. Dhariwal, "Glow: Generative flow with invertible 1×1 convolutions," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 10, 2018, pp. 215–224.
- [124] K. Chen, H. Zhou, H. Zhao, D. Chen, W. Zhang, and N. Yu, "When provably secure steganography meets generative models," 2018, *arXiv:1811.03732*. [Online]. Available: <http://arxiv.org/abs/1811.03732>



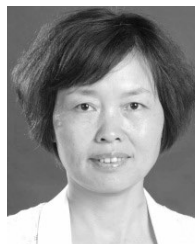
JIA LIU received the M.S. degree in cryptography from the Engineering University of PAP, Xi'an, China, in 2007, and the Ph.D. degree in pattern recognition and intelligent system from Shanghai Jiao Tong University, Shanghai, China, in 2012. He is currently an Associate Professor with the Key Laboratory of Network and Information Security, Engineering University of PAP. His research interests include machine learning and information security.



YAN KE received the B.S. degree in information research and security and the M.S. degree in cryptography from the Engineering University of Chinese People Armed Police Force (PAP), Xi'an, China, in 2014 and 2016, respectively, where he is currently pursuing the Ph.D. degree in cryptography. His research interests include information hiding, lattice cryptography, and deep learning.



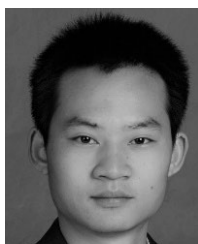
ZHUO ZHANG received the B.S. degree in information research and security from the Engineering University of People's Armed Police Force (PAP), Xi'an, China, in 2004, and the M.S. degree in cryptography from the National University of Defense Technology, China, in 2010. He is currently pursuing the Ph.D. degree in cryptography with the Rocket Force University of Engineering. He is currently an Assistant Professor with the Engineering University of PAP. His research interests include steganography and cryptography.



MINQING ZHANG received the M.S. degree in computer science and application and the Ph.D. degree in network and information security from Northwestern Polytechnical University, Xi'an, China, in 2001 and 2016, respectively. She is currently a Professor with the Key Laboratory of Network and Information Security, Chinese People Armed Police Force. Her research interests include cryptography and trusted computation.



YU LEI received the B.S. degree in information research and security and the M.S. degree in cryptography from the Engineering University of PAP, Xi'an, China, in 2008 and 2011, respectively. His research interests include mathematics statistics and cryptography.



JUN LI received the B.S. degree in information research and security and the M.S. degree in cryptography from the Engineering University of PAP, Xi'an, China, in 2009 and 2012, respectively. His research interests include mathematics statistics and cryptography.



XIAOYUAN YANG received the B.S. degree in applied mathematics and the M.S. degree in cryptography and encoding theory from Xidian University, Xi'an, China, in 1982 and 1991, respectively. He is currently a Professor with the Key Laboratory of Network and Information Security, PAP. His research interests include cryptography and trusted computation.

...