Generative Image Steganography Based on Text-to-Image Multimodal Generative Model

Jingyuan Jiang, Zichi Wang, Zihan Yuan, Xinpeng Zhang

***Abstract*—Image steganography, the technique of hiding secret messages within images, has recently advanced with generative image steganography, which hides messages during image creation. However, current generative steganography methods often face criticism for their low extraction accuracy and poor robustness—particularly their vulnerability to JPEG compression. To address these challenges, we propose a novel generative image steganography method based on the text-to-image multimodal generative model (StegaMGM). StegaMGM utilizes the initial random normalization distribution in the generative process of latent diffusion models (LDMs), the secret message is hidden in the generated image through message sampling, ensuring it follows the same probability distribution as typical image generative. The content of the stego image can also be controlled through the prompts. On the receiver side, using the shared prompt and diffusion inversion, can extract secret message with high accuracy. In the experimental section, we conducted detailed experiments to demonstrate the advantages of our proposed StegaMGM framework in extraction accuracy, resistance to JPEG compression, and security.**

***Index Terms*—Generative Image Steganography, Text-to-Image, Multimodal Generative Model, LDMs, DPM-Solvers++**

I. INTRODUCTION

Steganography is a technique hides secret message within normal media to enable covert communication through public channels. When the cover is an image, it is called image steganography. We categorize it into two main types: embedding based steganography and steganography without embedding (SWE) ([18], [19], [20], [25], [26], [36]). The first type has an inherent risk: the modifications to the cover image leave traces that are especially noticeable in high-capacity steganography. Stego images containing these traces can be detected through advanced steganalysis techniques ([2], [40], [41]).

To prevent trace detection, a method known as steganography without embedding (SWE) ([22], [35]) has been introduced, demonstrating strong resistance to common steganalysis techniques. Various generative models, such as generative adversarial networks, (GANs), normalizing flow, and variational autoencoders (VAEs), have shown remarkable advancements in generating images and have been utilized for image steganography. These models hide secret messages through mapping functions between latent space and secret message.

In spite of the progress made with generative steganography, there are still a number of drawbacks that need to be resolved. In terms of visual realism, the DCGAN-Steg image [15] and the autoregressive model-based image [39] do not demonstrate sufficient realism. The quality of steganographic images is enhanced by the image disentanglement autoencoder for steganography (IDEAS) [23] and the generative steganography network (GSN) [35]. Despite the fact that secret messages are still converted into images inefficiently, secret message extraction remains limited in accuracy. In addition, S2IRT [26] causes a disturbance in the distribution during sampling of latent space because it uses the Glow model for its mapping function. Compared to normal images, stego images can be easily distinguished by this disturbance. StegaDDPM [1] was the first to introduce diffusion models into steganography related work, but it is a model based on naive DDPM, as a Stochastic Differential Equation (SDE) Diffuser model, its performance is limited, and the efficiency of image generative is low.

To tackle the aforementioned challenges and elevate the effectiveness of image steganography, we explored the highly successful multimodal generative models for text-to-image synthesis: LDMs [9]. This model has shown exceptional performance in the field of image generative. LDMs [9] can generate desired content based on the input prompt and random Gaussian noise. The model is primarily divided into two parts: Contrastive Language-Image Pre-training (CLIP) [10]and the diffusion model. The CLIP [10] component converts the prompt into an embedding vector, while the diffusion model is responsible for transforming random Gaussian noise into an image. The embedding vectors help the model grasp the meaning of the prompt and guide the diffusion model to generate corresponding images based on the prompt. DDPM [4], [6], [12] and Denoising Diffusion Implicit Models (DDIM) [5] are commonly used diffusion models.

We propose a framework called StegaMGM. StegaMGM utilizing LDMs [9] as the foundation and introducing Diffusion probabilistic model-solver++ (DPM-Solver++) [3] to replace the original DDPM [4], [6], [12] or DDIM [5] as the image generative diffusion model. During the inversion of sampling, we employ the Inv-DPM [7] method to extract secret message accurately. As illustrated in Fig.1, this is a flowchart of generative

This work was supported in part by Natural Science Foundation of China under Grant U22B2047, 62376148, and supported in part by the Chenguang Program of Shanghai Education Development Foundation and Shanghai Municipal Education Commission under Grant 22CGA46.

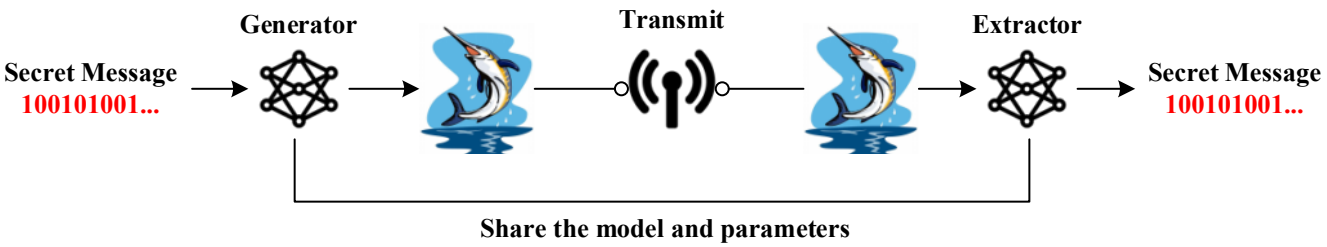


Fig. 1. Generative steganography involves a process where secret message is transformed into natural looking images through a generator. These images are then sent across lossless channels. An extractor is used at the receiving end to extract the secret message from the received images.

steganography, our approach also does not rely on a cover image, instead, refer to the method used by[48], [49], we design an algorithm for the bidirectional mapping of secret message to normal distributed noise. By using the obtained Gaussian noise and the input prompt, we can generate the stego image.

Compared to existing methods, our proposed StegaMGM is a steganography scheme based on a text-to-image multimodal generative model, allowing the content of the stego image to be controlled via prompts. The resulting stego image has extremely high quality and diversity. Stego image can be directly achieved without any additional training or fine-tuning for models. Additionally, since we map the binary bitstream of the secret message directly to the standard normal distribution of the latent variable z used for generating images, there is no need for any form of secret message embedding or cover image. The process of extracting the secret information is also simple and efficient. Furthermore, this scheme possesses a certain degree of resistance to JPEG compression.

The main contributions of this paper can be summarized as follows:

- We propose an image steganography model based on a multimodal generative model it is called StegaMGM. We also design a method that directly derives standard normal distributions from the secret message. Through this method, we can generate arbitrary stego images and completely break free from the constraints of cover images. The visual quality of the stego images is also exceptionally high. Experiments have demonstrated that the probability distribution of the stego images closely resembles that of the normal images;
- The model we propose has achieved a very high level of accuracy in secret message extraction. Experimental results indicate that the accuracy of secret message extraction exceeds 99%, and after the stego images undergo JPEG compression, the accuracy of secret message extraction can reach over 96%;
- Because the generative of the stego images are controlled by prompts and through the DPM solver++ sampling, the content of stego image can be controlled and their security is guaranteed;

II. Related Work

A. Steganography Related Work

In the field of steganography, secret messages are hidden inside cover media for the purpose of converting communications [8]. Steganography has been investigated using several types of cover media, including digital audio, digital images, digital videos, and text, with images being the most popular.

Prior research [25] had embedded secret messages in cover images by modifying the least significant bits of each pixel, but this method was susceptible to statistical steganalysis. To improve security, the first adaptive steganography method, known as highly undetectable stego (HUGO) [26], was developed. In spite of this, some adaptations of HUGO were made specifically for image areas with smooth edges. Subsequently, in 2012, the wavelet obtained weight (WOW) [13] method was developed to embed data exclusively in regions with significant texture or noise. [21] propose a novel image steganography framework that is robust for communication channels offered by various social networks. [11] proposed a technique that hides information into multiple covers. In contrast to the previously mentioned techniques, Baluja et al. [28] developed an advanced neural network to conceal a secret color image within a cover image. Thereafter, invertible neural networks (INNs) [16], [24], and [37] have been exploited for steganographic applications [14], exhibiting impressive results. Acknowledging that steganography reliant on embedding invariably leaves discernible alterations on the cover image, the perennial struggle against steganalysis remains unresolved.

With steganography without embedding, the cover image is not modified in any way to conceal a secret message. A new image SWE method based on GANs has been introduced by Hu et al. [15], [42], which maps the secret message into a latent vector that is fed to a generator in order to produce the stego image and messages are extracted by the extractor, which is trained to do so. Although this approach offers certain benefits, it also faces notable challenges, including a high rate of decoding errors, restricted steganographic capacity, and subpar visual quality. [43] proposed a GAN-based steganography work, which has good robustness and can generate stego images with high visual quality. Zhang et al. [46] transform secret messages

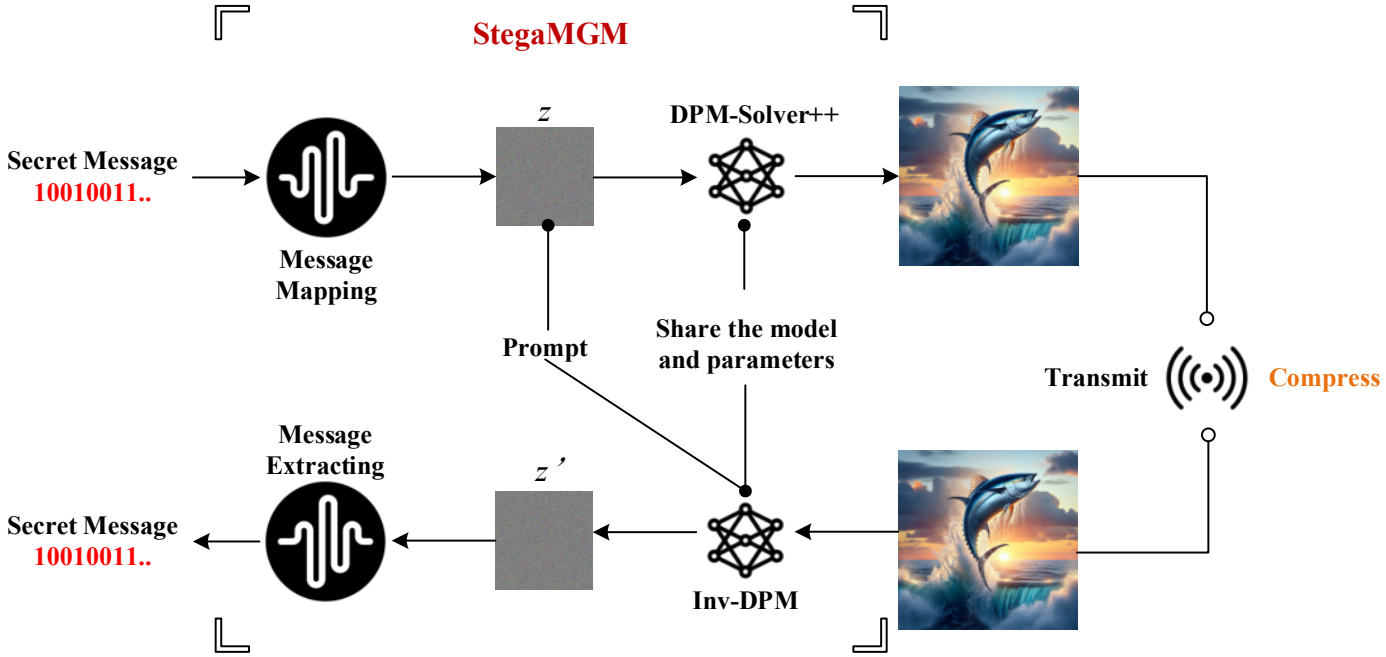


Fig. 2. Architecture of the proposed StegaMGM. Firstly, message sender performs a message mapping process on the binary bitstream of secret message, resulting in Gaussian noise z following a standard normal distribution. Subsequently, the generated Gaussian noise is utilized as the initial latent variable for DPM-Solver++, combined with the prompt for sampling, leading to the stego image. After the stego image is transmitted through the network channel, the received image at the receiver's end may experience some loss or compression. The receiver first performs a diffusion inversion using the prompt and the same model and parameters to obtain the corresponding latent variable z' at the initial moment. Then, the latent variable z' undergoes the message extraction process to retrieve the original secret message.

into class labels and integrate them into the GAN image generation process. Similarly, You et al. [23] transform secret messages into semantic information, such as facial expressions, within generated images, streamlining the creation of compact sticker images. In Yang et al. [39], they used autoregressive models to generate images, yet this also resulted in poor image quality due to autoregressive models' inherent limitations. Liu and his team, as early as 2022, introduced an autoencoder for image disentanglement (IDEAS) [23] that leveraged structural representation to enhance secret message decoding. Although this method was efficient, there were issues with irreversibility in the secret-to-image transformation. In order to address this challenge, Zhou et al. [26] introduced a method named S2IRT, where the Glow model is employed to transform the latent space into the image space, which has a complex distribution, by utilizing the multivariate normal distribution for the latent space. Despite this innovation, S2IRT's visual quality and diversity remain limitations. Furthermore, disruptions in latent space distribution compromise security.

StegaDDPM [1], a framework used diffusion model into steganography mask, but it did not consider the issues of multi-modality and JPEG compression. Additionally, Denoising Diffusion Probabilistic Models (DDPM) [4], [6], [12] as a SDE Diffuser, its potential for further development in the field of steganography is quite limited. [44] proposed a steganographic model based on LDM, which has good robustness and practicality. [45] proposed a steganography scheme based on LDMs using the ODE Diffusion method. However, the secret message hiding capacity of the above mentioned methods are limited.

B. Text to Image Multimodal Generative Model

In the field of Computer Vision (CV), pre-trained models are still primarily trained using manually annotated data. Due to the substantial workload involved in manual annotation, many scientists have started to explore more efficient and convenient methods for training visual representation models. In 2021, the CLIP [10] model was proposed, illustrating how to train transferable visual models using supervision signals from natural language processing (NLP). Because of its integration with NLP, the visual features learned by CLIP [10] have formed a strong association with the language used to describe specific objects. Consequently, this text-image paired pre-training model, CLIP [10], has had a significant impact on subsequent generative AI model, heralding a grand era in the field of Text-to-Image generative.

In 2022, [17] proposed DALL-E-2. In summary, DALL-E-2 trained three models to achieve Text-to-Image generative. The CLIP model is responsible for linking text and visual images, the GLIDE [28] model is responsible for generating images from visual descriptions, and the PRIOR model is responsible for mapping textual descriptions to visual descriptions.

LDMs [9] from Stability AI is a system composed of multiple components and models. Firstly, the text encoder is actually CLIP [10], which takes the input text and outputs an embedding vector representing each word/token in the text. This

Algorithm 1 Message Mapping Function D

Input Secret Message $m = \{m_1, m_2, \dots, m_{p\times n}\}$, where $m_i \in \{0,1\}$ (binary bitstream)

Output latent variable $z = \{z_1, z_2, \dots, z_n\}$

1. Divide the entire real number domain $(-\infty, \infty)$ into 2^p distinct, $q_k = \Phi^{-1}(k/p), k = (0,1,2,\dots,p)$
2. According to the value of each bit or the value of p consecutive bits, sampling is performed in the corresponding equal probability interval.
3. Get a sequence of length n hich follows a standard normal distribution, it becomes z

information is then presented to the image generator, in the latent space, comprises a UNet and a scheduling algorithm, iteratively generating images, typically using DDPM [4], [6], [12] or DDIM [5], but other more effective models can be substituted. The image decoder then renders the images based on the information obtained from the diffusion model. LDMs have generated many variants, achieving significant success in various fields and attracting attention from researchers.

In summary, text-to-image multimodal generative model have garnered significant attention and success across various fields, and their applications in the field of steganography also hold significant potential.

III. Proposed Method StegaMGM

In this paper, we propose StegaMGM, a multimodal, cover image-independent model for information hiding, which generates high-quality images and is robust against JPEG compression. The architecture of this model is presented in Fig. 2.

A. Message Mapping

In our proposed scheme, the binary bitstream of secret message is first converted into a tensor composed of random numbers that follow a standard normal distribution, which serves as the initial latent variable z for the generative model.

Initially, we employ the inverse function of the cumulative distribution function (CDF) associated with the normal distribution to partition the standard normal distribution into 2^p intervals. As shown in Figure 2. Subsequently, the secret data is sampled to the appropriate interval:

$$\forall t \in \{0,1,\dots,n-1\}, \exists!\, \mathcal{L}_{B_t} \subseteq \mathbb{R}$$
$$\text{s.t.} \quad \begin{cases} s_t \sim \mathcal{N}(0,1)|_{\mathcal{L}_{B_t}} \\ \mathcal{L}_{B_t} = [(\Phi^{-1}(\frac{dec(B_t)}{2^p}), \Phi^{-1}(\frac{dec(B_t)}{2^p})] \end{cases} \quad (1)$$

In Equation. 1:

- $m = \{m_1, m_2, \dots, m_{p\times n}\}$ is the secret message binary stream of length $p \times n$;
- $B_t = (m_{tp+1}, \dots, m_{(t+1)p})$ is the t-th p-bit block ($t \in [0, n-1]$);
- $\mathcal{L}_k$ is the equiprobable intervals of the standard normal distribution, where $\Phi(\mathcal{L}_k) = 2^{-p}$;
- $dec(B_t)$ is the decimal conversion of the p-bit binary block B_t;
- $|_{\mathcal{L}_{B_t}}$ means sampling constrained to interval $\mathcal{L}_{B_t}$;

Although the sampling is limited to a specific interval, but the latent variable $z = \{z_1, z_2, \dots, z_n\}$ still maintains the standard normal distribution.

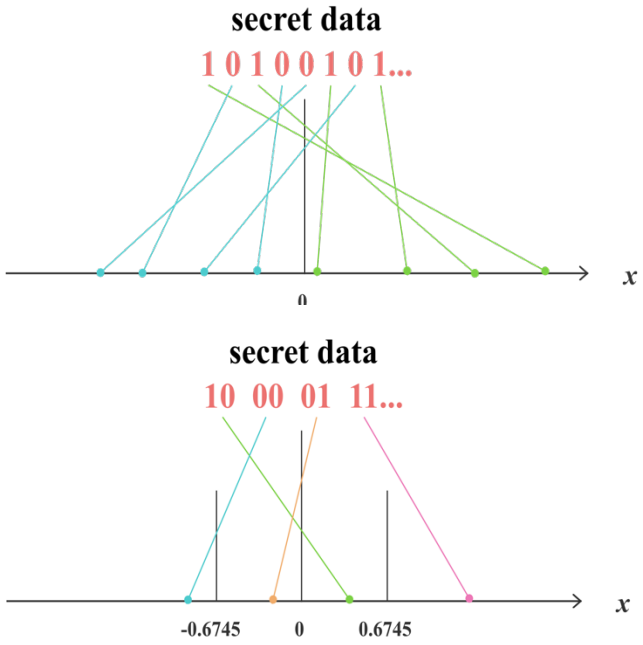


Fig. 3. In the first figure when p=1, we partition the standard normal distribution at zero, sampling the 0 of the secret data to the right segment and the 1 to the left segment. The second figure shows that when p=2, we divide the secret data into pairs (00, 01, 10 and 11) and sample them into four different regions of the standard normal distribution.

This method enables the efficient transformation between secret data and a standard normal distribution, and it is pertains to the design in [48],[49], but differs in that it employs the numerical values of secret data directly for sampling and we optimizing the sampling process, thereby enhancing efficiency. Details of the message mapping process are given in Algorithm 1.

B. Stego Image Generation Process

Upon obtaining the latent variable z, it is subsequently utilized to generate the stego image. StegaMGM has made significant progress in generating stego images compared to previous work. Both the quality of the stego images and the efficiency of image generative have reached a high level. Additionally, our improved model based on LDMs [9] allows for control of the content of the generated stego images through prompts, which was not achieved in previous steganography-related work. Moreover, the prompt can serve as a key to protect the stego image, ensuring that the secret message can only be accurately extracted by the receiver, who knows the prompt used to generate the stego image, therefore, even if the stego image and the model are intercepted, it still maintains a certain level of security. Additionally, because LDMs are multimodal generative models based on diffusion models—which essentially act as Gaussian denoisers and inherently possess robustness to noise and perturbations—we can still extract the secret information with high accuracy, even if the stego image is degraded during transmission.

The original LDMs [9] consists of CLIP [10] and DDPM [4],

[6], [12] or DDIM [5], but in our experiments, we found that neither DDPM [4], [6], [12] nor DDIM [5] could achieve our desired results. Upon further investigation, DPM-Solver++ [3]piqued our interest. Because it is essential to consider not only the quality of the generated stego image but also the accuracy during the secret message extraction stage, choosing the correct diffusion model is crucial for the performance of our proposed model.

These DPMs mentioned above are a class of diffusion models that generate original clean data through iterative denoising. The repeated noise reduction in DPMs usually demands extended sampling durations, and overcoming this challenge has been a focus of ongoing research. In DDIM [5] and DPM-Solver++ [3], the denoising process of DPM is formulated as an ordinary differential equation (ODE) and subsequently solved using methods such as the forward Euler method or exponential integrators. This method cuts down the necessary sampling steps from 1000 to as few as 50 or even 10. This progress not only significantly increases the efficiency of image generative but also provides theoretical support for the inversion of sampling, that is important for secret message extraction. These training-free methods are also practical for use with open-source DPMs.

The DPM-solver++ [3] is designed to recover $x_0 \in \mathbb{R}^D$ (stego image in our scenario) from $x_T \in \mathbb{R}^D$ (latent variable z in our scenario), which is considered to have undergone the following diffusion process (gradually adding Gaussian noise) defined in $t \in [0, T]$:

$$q_{t0}\left(x_t \middle| x_0\right) = \mathrm{N}\left(x_t; \alpha_t x_0, \sigma_t^2 \boldsymbol{I}\right) \tag{2}$$

where α_t^2 / σ_t^2, denotes the signal-to-noise ratio (SNR), which decreases strictly as t increases [34]. It is possible to perform sampling x_0 by solving the diffusion ODE, as follows:

$$\frac{dx_t}{dt} = \left(f(t) + \frac{g^2(t)}{2\sigma_t^2} \right) - \frac{\alpha_t g^2(t)}{2\sigma_t^2} x_\theta\left(x_t, t\right) \tag{3}$$

where $x_T \sim N(0, \tilde{\sigma}^2 I)$, $g^2(t) = \frac{d\sigma_t^2}{dt} - 2\frac{d\log\alpha_t}{dt}\sigma_t^2$, $f(t) = \frac{d\log\alpha_t}{dt}$ [34]. $x_\theta(x_t, t)$ is data prediction model parameterized by learnable θ, which aims to estimate x_0 from x_t

Lu et al. [38] have shown that ODE solvers utilizing exponential integrators achieve notably faster convergence than traditional solvers when dealing with Equation (3). With respect to a given initial value at time $s > 0$, DPM-Solver++ [3] computed the solution x_s for the diffusion ODE (Equation. (3)) at time t by using an exponential integrator:

$$x_t = \frac{\sigma_t}{\sigma_s} x_s + \sigma_t \int_{\lambda_s}^{\lambda_t} e^{\lambda} x_\theta\left(x_\lambda, \lambda\right) d\lambda \tag{4}$$

In this case, $x_\lambda = x_{t_\lambda(\lambda)}$ represents the change in log-SNR(λ) variable form. $\lambda_t = log\left(\frac{\alpha_t}{\sigma_t}\right)$ is the inverse of $t_\lambda(\cdot)$.

By applying the Taylor series expansion at $\lambda_{t_{i-1}}$ DPM-Solver++ [3] estimates the precise solution at time t_i, provided $x_{t_{i-1}}$ at time t_{i-1}:

$$x_{t_i} = \frac{\sigma_{t_i}}{\sigma_{t_{i-1}}} x_{t_{i-1}} + \sigma_{t_i} \sum_{n=0}^{k-1} \underbrace{x_\theta^{(n)}\left(x_{\lambda_{t_{i-1}}}, \lambda_{t_{i-1}}\right)}_{\text{estimated}} \underbrace{\int_{\lambda_{t_{i-1}}}^{\lambda_{t_i}} e^{\lambda} \frac{\left(\lambda - \lambda_{t_{i-1}}\right)^n}{n!} d\lambda}_{\text{analytically computed}} + \underbrace{\mathrm{O}\left(h_i^{k+1}\right)}_{\text{omitted}} \tag{5}$$

where $h_i = \lambda_{t_i} - \lambda_{t_{i-1}}$. $\mathcal{O}\left(h_i^{k+1}\right)$ can be omitted and the integral part can be computed analytically. So $x_\theta^{(n)}\left(x_{\lambda_{t_{i-1}}}, \lambda_{t_{i-1}}\right)$ for $n = 0, \ldots, k$ is the only part we need to find. The most straightforward estimate is $k = 1$, which corresponds to DDIM.

$$x_{t_i} = \frac{\sigma_{t_i}}{\sigma_{t_{i-1}}} x_{t_{i-1}} - \alpha_{t_i}\left(e^{-h_i} - 1\right) x_\theta\left(x_{t_{i-1}}, t_{i-1}\right) \tag{6}$$

Choosing $k = 2$ will result in a more precise approximation (and fewer steps):

$$x_{t_i} = \frac{\sigma_{t_i}}{\sigma_{t_{i-1}}} x_{t_{i-1}} - \alpha_{t_i}\left(e^{-h_i} - 1\right) \left(\left(1 + \frac{1}{2r_i}\right) x_\theta\left(x_{t_{i-1}}, t_{i-1}\right) - \frac{1}{2r_i} x_\theta\left(x_{t_{i-2}}, t_{i-2}\right) \right) \tag{7}$$

This is the one step of generate process of DPM-Solver++ (2M) we used, where '2M' indicates second-order multistep, in the course of the experiment, the generative of a stego image necessitates only ten steps.

C. Secret Message Extraction Process

In our proposed method, StegaMGM, the process of extracting the secret message is essentially the inverse of the stego image generative process, as shown in Figure 4. Firstly, after the receiver obtains the stego image, regardless of whether it has undergone JPEG compression, the inversion of image generative sampling is performed. In our proposed model, we adopt the Inv-DPM [7] scheme as the inversion of image generative sampling. After the stego image undergoes the diffusion inversion, the noise z' that approximates the noise latent variable z used to generate the stego image can be obtained. After obtaining the approximate noise z', the binary secret message bitstream can be obtained using the message extract process.

In the aforementioned process, the most crucial part is the diffusion inversion of the stego image generative. Inv-DPM, which plays a significant role in both the accuracy of secret message extraction and the performance against JPEG compression.

Before we talk about Inv-DPM [7], we need talk about naïve DDIM inversion first. The inversion of DDIM implies the obtaining $x_{t_{i-1}}$ given x_{t_i} (in our scenario, from stego image to latent variable), so $x_\theta(x_{t_i}, t_{i-1})$ as in Equation. (6) is not explicitly obtainable. The naïve DDIM inversion uses $x_\theta(x_{t_i}, t_{i-1})$ rather than $x_\theta(x_{t_{i-1}}, t_{i-1})$ to avoid the computational overhead associated with the implicit method.

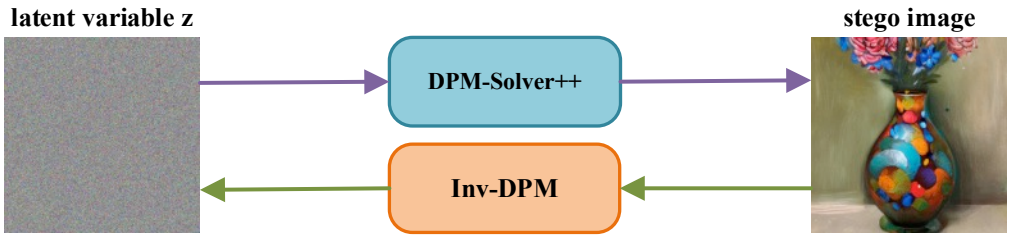


Fig. 4. The extraction of secret message is, in fact, the inverse procedure of stego image generation. The figure illustrates an ideal scenario where lossless conversion between secret information and the stego image is achieved through the generative model and its inverse process.

Algorithm 2 Inversion of DPM-Solver++(2M)

Require: initial value x_0, time steps $\{t_i\}_{i=0}^{M}$, data prediction model z_θ, UPDATE, $\mathcal{D}^\dagger$ (Decoder inversion)

Denote $h_i = \lambda_{t_i} - \lambda_{t_{i-1}}$ and $r_i = \frac{h_{i-1}}{h_i}$ for $i = 1, \ldots, M$.

$z'_{t_M} \leftarrow \mathcal{D}^\dagger(x_0)$ **if** LDMs **else** x_0

for $i \leftarrow M$ to 2 **do** $y'_{t_i} \leftarrow z'_{t_i}$

  for $j \leftarrow 1$ **to** $2J$ **do**

    $y'_{t_{i-j/J}} \leftarrow \frac{\sigma_{t_{i-j/J}}}{\sigma_{t_{i-(j-1)/J}}}(y'_{t_{i-j/J}} + \alpha_{t_{i-j/J}}(e^{-h_{i-j/J}} - 1) z_\theta(y'_{t_{i-(j-1)/J}}, t_{i-j/J}))$

  end for

  $z'_{t_{i-1}} \leftarrow y'_{t_{i-1}}$

  repeat

    $d'_i \leftarrow z_\theta(z'_{t_{i-1}}, t_{i-1}) + \frac{1}{2r_i}(z_\theta(y'_{t_{i-1}}, t_{i-1}) - z_\theta(y'_{t_{i-2}}, t_{i-2}))$

    $\hat{z}_{t_i} \leftarrow \frac{\sigma_{t_i}}{\sigma_{t_{i-1}}} z'_{t_{i-1}} - \alpha_{t_i}(e^{-h_i} - 1) d'_i$

    UPDATE $(z'_{t_{i-1}}; z'_{t_i}, \hat{z}_{t_i})$

  until converged

end for

$z'_{t_0} \leftarrow \frac{\sigma_{t_0}}{\sigma_{t_1}}(z'_{t_1} + \alpha_{t_1}(e^{-h_1} - 1) z_\theta(z'_{t_1}, t_0))$

repeat

  $\hat{z}_{t_0} \leftarrow \frac{\sigma_{t_1}}{\sigma_{t_0}} z'_{t_0} + \alpha_{t_1}(e^{-h_1} - 1) z_\theta(z'_{t_0}, t_0)$

  UPDATE$(z'_{t_0}; z'_{t_1}, \hat{z}_{t_1})$

until converged

return z'_{t_0}

The naïve DDIM inversion can be represented in the following manner, step by step:

$$x'_{t_{i-1}} = \frac{\sigma_{t_{i-1}}}{\sigma_{t_i}}\left(x_{t_i} + \alpha_{t_i}\left(e^{-h_i} - 1\right) x_\theta\left(x_{t_i}, t_{i-1}\right)\right) \tag{8}$$

This approach can be viewed as a variation of the forward Euler technique beginning at $t = 0$, effectively serving as the precise reverse of sampling with the backward Euler method. Because of values prior to t_{i-1} (*i.e.*, $x_{t_{i-2}}, x_{t_{i-3}}, \ldots$) cannot be estimated at the current time, and it have been used for higher-order terms in Equation. (6), *i.e.*,

$$\sigma_{t_i} \sum_{n=0}^{k-1} x_\theta^{(n)}\left(x_{\lambda_{t_{i-1}}}, \lambda_{t_{i-1}}\right) \int_{\lambda_{t_{i-1}}}^{\lambda_{t_i}} e^{\lambda} \frac{\left(\lambda - \lambda_{t_{i-1}}\right)^n}{n!} d\lambda \tag{9}$$

This scheme estimates these values (*i.e.*,$x_{t_{i-2}}$,$x_{t_{i-3}}$,…) by using a slightly less accurate method, like a naive DDIM inversion with a finer step size, and find that $x'_{t_{i-1}}$ by the backward Euler method as the high-order terms (Equation. (8)) are treated as constant.

Here we give you Algorithm 2 to illustrate this scheme with inversion of DPM-Solver++(2M) (Equation. (7)). In practical applications, no additional data transmission is required.

Firstly, the scheme obtain the substitutes of $x'_{t_{i-1}}$ and $x'_{t_{i-2}}$, $y'_{t_{i-1}}$ and $y'_{t_{i-2}}$, by using a fine-grained naïve DDIM inversion.

Then using $y'_{t_{i-1}}$ and $y'_{t_{i-2}}$, find $x'_{t_{i-1}}$ via the backward Euler method with high-order term approximation as follows:

$$d'_i \leftarrow z_\theta\left(z'_{t_{i-1}}, t_{i-1}\right) + \underbrace{\frac{z_\theta\left(y'_{t_{i-1}}, t_{i-1}\right) - z_\theta\left(y'_{t_{i-2}}, t_{i-2}\right)}{2r_i}}_{\text{high-order term approximation}} \tag{10}$$

where: $r_i = \frac{\lambda_{t_{i-1}} - \lambda_{t_{i-2}}}{\lambda_{t_i} - \lambda_{t_{i-1}}}$, and these operations just mentioned are repeated until convergence is achieved, and z_θ is a data prediction model that shares the same parameters and architecture as x_θ. However, x_θ utilizes forward Euler during the generative process, while z_θ employs backward Euler during inversion. To differentiate them, we refer to it as z_θ here.

Decoder inversion: Due to the fact that LDMs use latent variables in their sampling process, they must use a decoder($\mathcal{D}$) in order to convert the latent variable(z) into the image(x_0), and in reconstruction research, encode ($\mathscr{E}$) is equivalent to inversion of decoding. Nevertheless, the encoder does not represent the decoder exactly, so reconstruction errors occur. To reduce errors, the scheme performs an exact inversion of the decoder, and, as with many GAN inversion studies, it uses gradient descent as follows:

accuracy of our model after the stego image underwent JPEG compression. The results are shown in Table I. The results indicate that our proposed method not only exhibits extremely

Fig. 5. The pictures generated by our proposed StegaMGM are shown above, the first row and the third row are normal images, and the second row and the fourth row are stego images. It is obvious that the content of two pairs of images in each column are highly relevant, because we use the same prompt to control the image content when generating images. It can be seen that whether it is generating normal images or stego images, StegaMGM has good image generative quality.

1: **function** $\mathrm{D}^{\dagger}(x)$ *// Decoder inversion*
2: $z \leftarrow \mathrm{E}(x)$
3: **repeat** gradient step on $\nabla_z \| x - \mathrm{D}(z) \|_2^2$
4: **until** converged
5: **return** z
6: **end function**

By employing Algorithm 2, it is possible to invert the stego image, thereby retrieving the latent variable z', approximate latent variable z. In the latent variable z', each value is sampled utilizing the message mapping method, so the hidden secret message can be confirmed based on the interval in which its value lies. For example, when p=1, if the value is negative, the hidden secret message is 0, and if the value is positive, the hidden secret message is 1.

IV. Experiments

Our experiment was conducted on a GeForce RTX 3090 GPU card, and it was not necessary to provide any additional training or fine-tuning to the diffusion model. In our experiment, we chose Stable Diffusion v2.1 as the base diffusion model. We set the guidance scale of Stable Diffusion to 3.0 in order to achieve invertible image translation.

To assess the precision of secret extraction, the visual quality of the generated stego images, and the security of our method, we utilize extraction accuracy (Acc), Frechet inception distance (Fid) [32], and detection error (Pe) respectively. Acc is calculated as:

$$\mathrm{Acc} = \frac{d \odot d'}{len(d)} \tag{11}$$

where, and are the input secret message and extract secret message, is the element wise XNOR operation. Frechet inception distance (Fid) is a metric used to evaluate the quality and diversity of images generated by generative models. FID measures the distance between feature representations of real and synthetic images using a pre-trained inception network. A lower FID score indicates a higher level of quality and diversity in the images generated. Pe is a widely used metric for assessing the invisibility of steganographic images, defined as:

$$\mathrm{Pe} = \min_{p_{FA}} \frac{1}{2}(P_{FA} + P_{MD}) \tag{12}$$

There are two metrics, P_{FA} and P_{MD}, of false alarms rate and missed detections rate, respectively. Pe is a parameter that ranges between [0,1], and its optimal value is 0.5. When Pe is 0.5, the steganalysis tool is ineffective. Every generated image, except those specifically used for anti-JPEG compression testing, is saved as a PNG.

A. Accuracy

The accuracy of secret message extraction is a crucial metric for evaluating a steganography method. We tested the accuracy (Acc) of our proposed model StegaMGM, as well as the accuracy of our model after the stego image underwent JPEG compression. The results are shown in Table I. The results indicate that our proposed method not only exhibits extremely high accuracy in normal conditions but also maintains a robust accuracy even after the stego image undergoes JPEG

compression. Our proposed method can consistently achieve an accuracy of over 99%. In our experiments, we can further improve the accuracy to reach close to 100%, by utilizing error correction codes, albeit at the expense of sacrificing some payload. In Table I, '/' indicates that the authors did not consider the issue of facing JPEG compression. Considering the widespread practice of JPEG compression for transmitting images over networks, methods that do not take this factor into account lack practicality. Furthermore, we conducted experiments to evaluate the payload capacity of our proposed method. In our experiments, the dimensions of the generated single image are 512*512 pixels, while the latent variable tensor is characterized by dimensions of 1*4*64*64. So when p=1 in the message mapping method, the payload capacity of the proposed method is 1/16, the generated image conceals 16,384 bits of secret message. Additionally, we evaluated the accuracy of secret message extraction across varying payload capacities. The results are shown in Table II.

TABLE I
COMPARISON OF STEGAMGM WITH SOTA METHODS

Methods	Image size	Acc		
		PNG	JPEG q=80	JPEG q=50
StegaDDPM[1]	256*256	92.45%	/	/
IDEAS [23]	256*256	99.05%	/	/
S2IRT [26]	64*64	98.47%	59.31%	53.48%
LDStega[44]	256*256	98.65%	96.15%	94.29%
Ours	512*512	99.63%	96.23%	94.33%

TABLE II
PAYLOAD CAPACITY EXPERIMENT

payload	BPP	Capacity (bits)	Acc		
			PNG	JPEG q=80	JPEG q=50
p=1	1/16	16384	99.63%	96.23%	94.33%
p=2	1/8	32768	99.25%	94.38%	92.18%
p=3	3/16	49152	99.02%	91.67%	90.62%

TABLE III
ABLATION EXPERIMENTS OF STEGAMGM

Methods	Image size	Acc		
		PNG	JPEG q=80	JPEG q=50
DDIM [5] & naïve DDIM	512*512	95.33%	89.69%	85.31%
Ours	512*512	99.63%	96.23%	94.33%

Since JPEG compression primarily affects certain aspects of the image rather than the pixel values as a whole, only a small portion of pixel values are altered after compression. Consequently, this has minimal impact on the inversion of the diffusion model. Additionally, our designed message mapping process and message extraction process inherently possess some fault tolerance. Therefore, our proposed scheme offers a degree of resistance to JPEG compression. We conducted ablation experiments using DDIM and naive DDIM inversion as comparisons, and the results are shown in Table III.

B. Security

The security of steganography encompasses resistance to steganalysis detection as well as behavioral security. We initially randomly generated 10,000 normal images using the proposed StegaMGM and simultaneously generated 10,000 stego images for testing purposes. Initially, we assessed the anti-steganalysis performance of StegaDDPM[1] , IDEAS [23], S2IRT [26], and our newly proposed StegaMGM utilizing advanced steganalysis networks SRNet [2], Ye-Net [40], and SiaStegNet [41]. The comparative experimental results are presented in Table IV.

StegaDDPM [1] achieves distribution preservation through meticulously designed mapping functions, thereby achieving a reliable anti-steganalysis ability. The IDEAS [23] methods embed secret message into noise as input for generating images. This approach poses a challenge for steganalysis detectors, as it becomes difficult to discern the difference between noise samples generated from the normal process and those mapped from secret data. S2IRT [26] adapts a multi-Gaussian distribution of the secret space to an intricate distribution of the

TABLE IV
COMPARISON OF THE ANTI-STEGANALYSIS ABILITY

Methods	SRNet[2]	Ye-Net [40]	SiaStegNet [41]
StegaDDPM [1]	50.31%	51.03%	52.01%
IDEAS [23]	51.78%	52.31%	53.01%
S2IRT [26]	99.23%	99.87%	99.44%
Ours	50.22%	50.04%	50.82%

image space. Nonetheless, its mapping function disturbs the latent space's distribution typically followed during standard image generative, resulting in weakened steganography security that can be easily detected by steganalysis networks. The proposed StegaMGM achieves anti-steganalysis by directly sampling a standard normal distribution by using the secret message.

According to the information-theoretic definition of steganographic security is provided by Cathin [33]. Assuming the cover is sampled from C with probability distribution P_C, and the steganographic technique produces a stego object with distribution P_S. The disparity between these distributions can be quantified through relative entropy:

$$D(P_C \parallel P_S) = \sum_{x \in C} P_C(x) \log \frac{P_C(x)}{P_S(x)} \tag{13}$$

When $D(P_C \parallel P_S) = 0$ the steganography system is considered perfectly secure.

In this paper, the generative processes of both normal images and stego images are generated by same model. At any given moment during the image generative process, the distribution of image follows a normal distribution. The only difference between the generative processes of normal image and stego image is the initial latent z. The initial latent z for normal images is directly randomly sampled from a standard normal distribution, whereas the initial latent z for stego images is obtained through a message mapping process from secret message, yet still follows a standard normal distribution. Therefore, the difference in initial latent z does not affect the probability distribution during the subsequent image generative process, so the distributions of the two images meet the $D(P_C \parallel P_S) = 0$ at any moment throughout the entire generative process.

C. Other performance of StegaMGM

As shown in Figure 5, the model we proposed, StegaMGM, has achieved a high level in terms of image quality, diversity of generated images, and control over content. Due to the introduction of multimodal generative models, the steganography task will break free from the dependence on cover images and any form of embedding secret message, truly achieving generative steganography. Furthermore, it has enhanced the efficiency of stego image generative.

FID measures the difference between the distribution of generated data and the real data distribution by calculating the Fréchet distance between the two distributions. We use FID to measure the distribution difference between the generated normal images and stego images, which provides a quantitative metric to evaluate the performance of the proposed model. We tested the FID score using 10,000 normal images and 10,000 stego images generated by StegaMGM. During testing, we utilized DDIM and DPM-Solver replaced the DPM-Solver++ for comparison and varying the number of function evaluations (NFE). The experimental results are presented in Table V. Since we use DPM-Solver++ as the generation model, it only takes ten steps or less to generate high-quality images when generating stego images, the DDPM or DDIM method usually requires 1000 and 50 steps respectively. In our experimental environment, it only takes less than ten seconds to generate a single stego image. The extraction of secret information takes more time, about two minutes. Consequently, the proposed method is capable of producing stego images more efficiently, both in terms of reduced time and diminished computational resource requirements.

TABLE V
COMPARISON OF THE FID

Methods	NFE	FID↓
DDIM [5]	10	11.01
DPM-Solver [38]	10	9.98
Ours	10	6.97

As far as we know, FID, as a metric for measuring the distance between the distributions of two sets of images, has not been widely used as a reference in steganography tasks. Nonetheless, we still believe it is an effective way to demonstrate the advantages of our proposed StegaMGM. GSN [35] previously used FID to test their approach, and the FID score for GSN [35] when generating 128x128 stego images was 13.29, which is higher than that of our proposed model.

V. CONCLUSION

In this paper, we propose StegaMGM, an image generation steganography model derived from an improved LDMs model. This approach does not require any additional training or fine-tuning. In our proposed scheme, there is no process of embedding secret message at any stage. Instead, it uses the binary bitstream of the secret message itself to sample a latent variable containing the secret message, which is then used to generate the stego image. This scheme achieves an accuracy rate of over 99% during secret message extraction, and even after JPEG compression, the accuracy remains above 96%. Experimental tests demonstrate that our scheme has strong anti-steganalysis capabilities and security. Moreover, our approach demonstrates superior image generative quality and efficiency compared to existing methods.

REFERENCES

[1] Y. Peng, D. Hu, Y. Wang, K. Chen, G. Pei, and W. Zhang, "Stegad-dpm: Generative image steganography based on denoising diffusion probabilistic model," in *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 7143–7151, 2023.

[2] M. Boroumand, M. Chen, and J. Fridrich, "Deep residual network for steganalysis of digital images," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 5, pp. 1181–1193, 2018.

[3] C. Lu, Y. Zhou, F. Bao, J. Chen, C. Li, and J. Zhu, "Dpm-solver++: Fast solver for guided sampling of diffusion probabilistic models," *arXiv preprint arXiv:2211.01095*, 2022.

[4] J. Choi, S. Kim, Y. Jeong, Y. Gwon, and S. Yoon, "Ilvr: Conditioning method for denoising diffusion probabilistic models," *arXiv preprint arXiv:2108.02938*, 2021.

[5] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020.

[6] F.-A. Croitoru, V. Hondru, R. T. Ionescu, and M. Shah, "Diffusion models in vision: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 9, pp. 10850–10869, 2023.

[7] S. Hong, K. Lee, S. Y. Jeon, H. Bae, and S. Y. Chun, "On exact inversion of dpm-solvers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7069–7078, 2024.

[8] T. Filler, J. Judas, and J. Fridrich, "Minimizing additive distortion in steganography using syndrome-trellis codes," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 920–935, 2011.

[9] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

[10] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., "Learning transferable visual models from natural language supervision," in *International conference on machine learning*, pp. 8748–8763, PMLR, 2021.

[11] N. Zhong, Z. Qian, Z. Wang, X. Zhang and X. Li, "Batch Steganography via Generative Network," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 1, pp. 88-97, 2021.

[12] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[13] V. Holub and J. Fridrich, "Designing steganographic distortion using directional filters," in *2012 IEEE International workshop on information forensics and security (WIFS)*, pp. 234–239, IEEE, 2012.

[14] X. Hu, Z. Fu, X. Zhang and Y. Chen, "Invisible and Steganalysis-Resistant Deep Image Hiding Based on One-Way Adversarial Invertible Networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 7, pp. 6128-6143, 2024.

[15] 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.

[16] J. Jing, X. Deng, M. Xu, J. Wang, and Z. Guan, "Hinet: Deep image hiding by invertible network," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4733–4742, 2021.

[17] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *arXiv preprint arXiv:2204.06125*, vol. 1, no. 2, p. 3, 2022.

[18] V. Kishore, X. Chen, Y. Wang, B. Li, and K. Q. Weinberger, "Fixed neural network steganography: Train the images, not the network," in *International Conference on Learning Representations*, 2021.

[19] X. Zhang, J. Long, Z. Wang, and H. Cheng, "Lossless and reversible data hiding in encrypted images with public-key cryptography," *IEEE transactions on circuits and systems for video technology*, vol. 26, no. 9, pp. 1622–1631, 2015.

[20] B. Li, M. Wang, X. Li, S. Tan, and J. Huang, "A strategy of clustering modification directions in spatial image steganography," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 9, pp. 1905–1917, 2015.

[21] J. Tao, S. Li, X. Zhang, and Z. Wang, "Towards robust image steganography," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 2, pp. 594–600, 2018.

[22] Q. Liu, X. Xiang, J. Qin, Y. Tan, J. Tan, and Y. Luo, "Coverless steganography based on image retrieval of densenet features and dwt sequence mapping," *Knowledge-Based Systems*, vol. 192, p. 105375, 2020.

[23] X. Liu, Z. Ma, J. Ma, J. Zhang, G. Schaefer, and H. Fang, "Image disentanglement autoencoder for steganography without embedding," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2303–2312, 2022.

[24] S.-P. Lu, R. Wang, T. Zhong, and P. L. Rosin, "Large-capacity image steganography based on invertible neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10816–10825, 2021.

[25] J. Mielikainen, "Lsb matching revisited," *IEEE signal processing letters*, vol. 13, no. 5, pp. 285–287, 2006.

[26] Z. Zhou, Y. Su, J. Li, K. Yu, Q. J. Wu, Z. Fu, and Y. Shi, "Secret-to-image reversible transformation for generative steganography," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 5, pp. 4118–4134, 2022.

[27] T. Pevn`y, T. Filler, and P. Bas, "Using high-dimensional image models to perform highly undetectable steganography," in *Information Hiding: 12th International Conference, IH 2010, Calgary, AB, Canada, June 28-30, 2010, Revised Selected Papers 12*, pp. 161–177, Springer, 2010.

[28] A. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. McGrew, I. Sutskever, and M. Chen, "Glide: Towards photorealistic image generation and editing with text-guided diffusion models," *arXiv preprint arXiv:2112.10741*, 2021.

[29] S. Baluja, "Hiding images in plain sight: Deep steganography," *Advances in neural information processing systems*, vol. 30, 2017.

[30] W. Tang, B. Li, M. Barni, J. Li, and J. Huang, "An automatic cost learning framework for image steganography using deep reinforcement learning," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 952–967, 2020.

[31] W. Tang, S. Tan, B. Li, and J. Huang, "Automatic steganographic distortion learning using a generative adversarial network," *IEEE Signal Processing Letters*, vol. 24, no. 10, pp. 1547–1551, 2017.

[32] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *Advances in neural information processing systems*, vol. 30, 2017.

[33] A. Van Den Oord, N. Kalchbrenner, and K. Kavukcuoglu, "Pixel recurrent neural networks," in *International conference on machine learning*, pp. 1747–1756, PMLR, 2016.

[34] D. Kingma, T. Salimans, B. Poole, and J. Ho, "Variational diffusion models," *Advances in neural information processing systems*, vol. 34, pp. 21696–21707, 2021.

[35] P. Wei, S. Li, X. Zhang, G. Luo, Z. Qian, and Q. Zhou, "Generative steganography network," in *Proceedings of the 30th ACM International Conference on Multimedia*, pp. 1621–1629, 2022.

[36] N. Wu, P. Shang, J. Fan, Z. Yang, W. Ma, and Z. Liu, "Research on coverless text steganography based on single bit rules," in *Journal of Physics: Conference Series*, vol. 1237, p. 022077, IOP Publishing, 2019.

[37] Y. Xu, C. Mou, Y. Hu, J. Xie, and J. Zhang, "Robust invertible image steganography," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 7875–7884, 2022.

[38] C. Lu, Y. Zhou, F. Bao, J. Chen, C. Li, and J. Zhu, "Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps," *Advances in Neural Information Processing Systems*, vol. 35, pp. 5775–5787, 2022.

[39] K. Yang, K. Chen, W. Zhang, and N. Yu, "Provably secure generative steganography based on autoregressive model," in *International Workshop on Digital Watermarking*, pp. 55–68, Springer, 2018.

[40] J. Ye, J. Ni, and Y. Yi, "Deep learning hierarchical representations for image steganalysis," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 11, pp. 2545–2557, 2017.

[41] W. You, H. Zhang, and X. Zhao, "A siamese cnn for image steganalysis," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 291–306, 2020.

[42] C. Yu, D. Hu, S. Zheng, W. Jiang, M. Li, and Z.-q. Zhao, "An improved steganography without embedding based on attention gan," *Peer-to-Peer Networking and Applications*, vol. 14, no. 3, pp. 1446–1457, 2021.

[43] Z. Yang, K. Chen, K. Zeng, W. Zhang, and N. Yu, "Provably secure robust image steganography," *IEEE Transactions on Multimedia*, 2023.

[44] Y. Peng, Y. Wang, D. Hu, K. Chen, X. Rong, and W. Zhang, "Ldstega: Practical and robust generative image steganography based on latent diffusion models," in *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 3001–3009, 2024.

[45] X. Hu, S. Li, Q. Ying, W. Peng, X. Zhang, and Z. Qian, "Establishing robust generative image steganography via popular stable diffusion," *IEEE Transactions on Information Forensics and Security*, 2024.

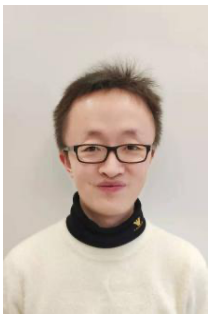
[46] Z. Zhang, G. Fu, R. Ni, J. Liu, and X. Yang, "A generative method for steganography by cover synthesis with auxiliary semantics," *Tsinghua Science and Technology, vol. 25, no. 4*, pp. 516–527, 2020.

[47] Z. You, Q. Ying, S. Li, Z. Qian, and X. Zhang, "Image generation network for covert transmission in online social network," in *Proceedings ofhe 30th ACM International Conference on Multimedia*, pp. 2834–2842, 2022

[48] K. Chen, H. Zhou, H. Zhao, D. Chen, W. Zhang, and N. Yu,"Distribution-preserving steganography based on text-to-speech generative models," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 5, pp. 3343–3356, 2021.

[49] Z. Yang, K. Zeng, K. Chen, H. Fang, W. Zhang, and N. Yu, "Gaussianshading: Provable performance-lossless image watermarking for diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12162–12171, June 2024.

Jingyuan Jiang received B.S. from Yanbian University, China, in 2017, and received the MS degree from Yanbian University, China, in 2019. He is currently pursuing the PhD degree in information and communication engineering in Shanghai University, China. His resea-rch interests include steganography and deep learning.
Email: jingjingyuan@shu.edu.cn



Zichi Wang received the BS degree in electronics and information engineering from Shanghai University, China, in 2014, and received the MS degree in signal and information processing in 2017, the PhD degree in information and communication engineering from the same university in 2020. His research interests include steganography, steganaly- sis, and artificial intelligence security. He has published over 50 papers in these areas.
Email: wangzichi@shu.edu.cn

Zihan Yuan received B.S. from Jining Medical University, China, in 2018, and received the MS degree from Ludong University, China, in 2021, She is currently pursuing the PhD degree in information and communication engineering in Shanghai University, China. Her research interests include information hiding and deep learning.
Email: imyuanzihan@163.com

Xinpeng Zhang received B.S. from Jilin University, China, in 1995, and the M.S. and Ph.D. from Shanghai University, in 2001 and 2004, respectively. Since 2004, he has been with the faculty of the School of Communication and Information Engineering, Shanghai University, where he is currently a fulltime Professor. He is also with the faculty of the School of Computer Science, Fudan University. He was with The State University of New York at Binghamton as a Visiting Scholar from 2010 to 2011, and also with Konstanz University as an experienced Researcher, sponsored by the Alexander von Humboldt Foundation from 2011 to 2012. His research interests include multimedia security, image processing, and digital forensics. He has published over 200 research papers. He was an Associate Editor for IEEE Transactions on Information Forensics and Security from 2014 to 2017.
Email: xzhang@shu.edu.cn