StegaStyleGAN: Towards Generic and Practical Generative Image Steganography

Wenkang Su[1, 2], **Jiangqun Ni**[1, 3*], **Yiyan Sun**[1]

[1]Sun Yat-Sen University
[2]Guangzhou University
[3]Peng Cheng Laboratory
swk1004@gzhu.edu.cn, issjqni@mail.sysu.edu.cn, sunyy27@mail2.sysu.edu.cn

Abstract

The recent advances in generative image steganography have drawn increasing attention due to their potential for provable security and bulk embedding capacity. However, existing generative steganographic schemes are usually tailored for specific tasks and are hardly applied to applications with practical constraints. To address this issue, this paper proposes a generic generative image steganography scheme called Steganography StyleGAN (StegaStyleGAN) that meets the practical objectives of security, capacity, and robustness within the same framework. In StegaStyleGAN, a novel Distribution-Preserving Secret Data Modulator (DP-SDM) is used to achieve provably secure generative image steganography by preserving the data distribution of the model inputs. Additionally, a generic and efficient Secret Data Extractor (SDE) is invented for accurate secret data extraction. By choosing whether to incorporate the Image Attack Simulator (IAS) during the training process, one can obtain two models with different parameters but the same structure (both generator and extractor) for lossless and lossy channel covert communication, namely StegaStyleGAN-Ls and StegaStyleGAN-Ly. Furthermore, by mating with GAN inversion, conditional generative steganography can be achieved as well. Experimental results demonstrate that, whether for lossless or lossy communication channels, the proposed StegaStyleGAN can significantly outperform the corresponding state-of-the-art schemes.

Introduction

Steganography (Fridrich 2009) is an important branch of information hiding that aims to conceal secret data in innocuous multimedia, including images, audio, video, text, etc. It is not only required to ensure that the secret data can be recovered exactly by the recipient but also to make the covert communication undetectable by the State-Of-The-Art (SOTA) steganalyzer (Fridrich and Kodovský 2012; Ye, Ni, and Yi 2017; Boroumand, Chen, and Fridrich 2019). Since digital images are most widely used in our daily lives, the majority of research on steganography has currently relied on images as cover.

Regarding the current conventional image steganography methods, the prevalent steganographic schemes (Holub, Fridrich, and Denemark 2014; Li et al. 2014; Su et al. 2018, 2020, 2022; Yang et al. 2020; Tang et al. 2021) usually embed secret data by modifying the cover image. Although these cover-modified steganographic schemes exhibit good resistance to steganalyzers, their steganographic capacities (Filler and Fridrich 2009) are still limited. To this end, some end-to-end neural network-based cover-modified approaches (Zhang et al. 2019a; Baluja 2020; Zhang et al. 2020; Ying et al. 2021; Jing et al. 2021; Lu et al. 2021) have recently been developed, showing huge steganographic capabilities. However, they are almost all Hiding-Images-withIn-Images (HI3) schemes, and secret images can only be recovered at the semantic level (i.e., similar in image content only). Moreover, compared to the conventional hand-crafted schemes, all these end-to-end neural network-based cover-modified approaches have an inherent flaw in that they are virtually invulnerable to detection by SOTA steganalyzers.

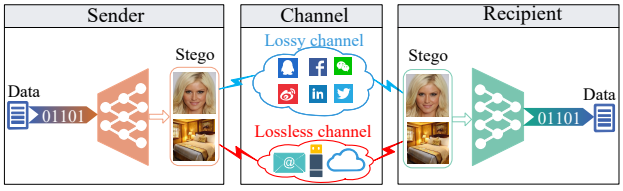

Figure 1: Sketch of our proposed generative image steganography scheme.

To improve resilience to steganalyzers, i.e., steganographic security, of the above deep learning-based schemes, and to enable recovery of secret data at the bit level (i.e., consistent at each bit), generative image steganography has been developed, which can automatically generate stego images according to secret data. Although generated samples were once considered "behavioral abnormal" from the perspective of covert communication, they are now taken for granted as generated media increasingly pervades our daily lives. In general, with given generative models, the generative steganography is empirically secure unless the generated samples are regarded as virtually created. This is because the generative schemes are cover-free in nature, which would prevent the existing steganalyzers (Fridrich and Kodovský 2012; Ye, Ni, and Yi 2017; Boroumand, Chen, and Fridrich 2019) designed for cover-modified steganography to distin-

*Corresponding author.

guish the generated stego from the generated cover. In this regard, Hu *et al.* (Hu et al. 2018) established a relationship between secret information and the input of DCGANs (i.e., the noise vector) to generate stego images, Wang *et al.* (Wang et al. 2018) used the concatenation of secret data and noise vector as input to generate the stego images, and Zhang *et al.* (Zhang et al. 2019b) embedded the secret data in advance to an uncorrupted region, which will be then fed into the trained DCGANs generator for semantic complementation to obtain the stego images. Furthermore, Hu *et al.* (Yu et al. 2021) utilized the attention method to improve the performance of their prior art (Hu et al. 2018). Although these algorithms have made notable progress, the resolution of the generated images is low and the image quality is still not very good. More importantly, the security guarantees of all these algorithms are based on the assumption that the Eavesdropper (who detects steganography) can not access the network model parameters (or even the model structure). And as reported in (Yu et al. 2021), once they are compromised, their security can hardly be guaranteed. To address these practical problems, Wei *et al.* (Wei et al. 2022) has recently presented an advanced generative steganographic network (GSN) that designs a modified StyleGAN2 (Karras et al. 2020) generator to generate high-security stego images, but with dramatic degradation in data extraction accuracy and image quality at large capacity. What's worse, all these methods are vulnerable to image attacks.

In addition to the conflict between security and capacity, robustness is another key issue to be discussed. Jessica pointed out early in (Fridrich 1999) that there is a triangular relationship between security, capacity, and robustness that constrains each other in information hiding. And for the study of information hiding robustness, some deep learning-based schemes have emerged recently. Dong *et al.* (Dong et al. 2021) devised a Facial Stego Image Synthesis method for data hiding with GAN (FSIS-GAN), which generates a realistic facial stego image (a.k.a. real semantics) from the secret data and key. But with the introduction of the image attack simulator, some speckle noises emerge in the generated stego images of FSIS-GAN. Li *et al.* (Li, Zhang, and Liu 2021) converted the secret data into the input latent of StyleGAN generator to directly generate stego images, whose capacity can reach 512 bits along with promising image quality, but the secret data extraction accuracy is only 50% on the test set due to the overfitting of the model. You *et al.* (You et al. 2022) mapped the secret data as latent vectors of controlling the adaptive instance normalization in StyleGAN (Karras, Laine, and Aila 2019) generator to generate the stego images. Although the improvement of stego image quality and high extraction accuracy of secret data can be achieved, the resolution of generated images is very low and the steganographic capacity is quite small. Again, the security guarantees for all these robust schemes are still based on the same assumptions aforementioned.

In a word, the above-mentioned schemes are usually custom-designed to meet only certain specific objectives and are hardly adequate for practical applications. In this paper, we propose a more generic and practical generative image steganography scheme based on StyleGAN2, namely StegaStyleGAN, dedicated to meeting the practical objectives of security, capacity, and robustness within the same framework. The sketch of our proposed scheme is shown in Figure 1, and the main contributions of this paper are summarized as follows:

- Design a novel Distribution-Preserving Secret Data Modulator (DP-SDM) to modulate the injected noise of StyleGAN2 generator with secret data, thus enabling visually convincing and provably secure steganography.
- By simply varying the training pipeline in the StegaStyleGAN scheme, two practical models, StegaStyleGAN-Ls and StegaStyleGAN-Ly, can be obtained for lossless and lossy channel covert communication, respectively.
- Develop an efficient and generalized Secret Data Extractor (SDE) with a densely connected structure, which can used for both StegaStyleGAN-Ls and StegaStyleGAN-Ly Models.
- The first to propose to combine GAN inversion to generate stego images with real semantics (existing in reality), facilitating generative steganography towards practical applications.

The Proposed StegaStyleGAN Scheme

Overview

As shown in Figure 2, our proposed StegaStyleGAN scheme includes a distribution-preserving secret data modulator (DP-SDM), a generator (G), a discriminator (D), an image attack simulator (IAS) and a secret data extractor (SDE). Unlike the previous art GSN (Wei et al. 2022), we do not modify the topology of StyleGAN2 and directly use StyleGAN2 generator and discriminator as G and D. As such, the pre-trained StyleGAN2 can be employed to accelerate the training of StegaStyleGAN. Moreover, instead of directly replacing the original injected noise of StyleGAN2 with secret data as in GSN, we fuse it with the injected noise through the DP-SDM, dedicated to keeping the noise distribution used to generate the stego and cover the same. Following this scheme, a large-capacity and secure generative steganographic model for lossless channel covert communication, namely **StegaStyleGAN-Ls**, can be directly obtained. In addition, by introducing the Image Attack Simulator (IAS), to assist the training, another robust and secure generative steganographic model for lossy channel covert communication, namely **StegaStyleGAN-Ly**, can be further obtained.

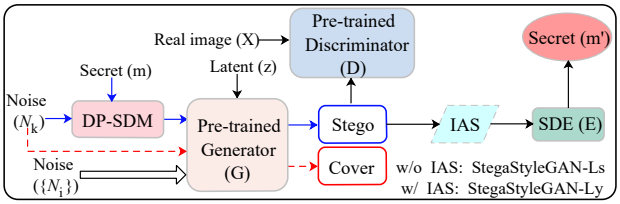

Figure 2: The framework of our proposed StegaStyleGAN scheme. The cover and stego images are generated by G when inputs ($\mathbf{z}$,$\{N\}$) (*dashed red arrow*) and ($\mathbf{z}$,$\{N\}$,$\mathbf{m}$) (*solid blue arrow*), respectively.

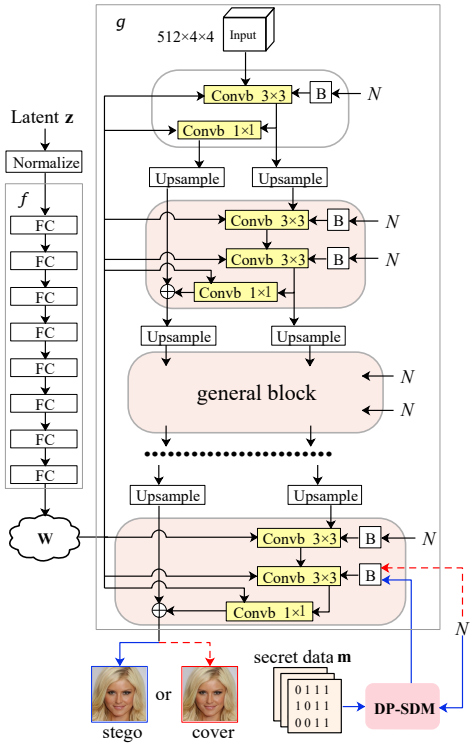

Figure 3: The generator architecture of our proposed StegaStyleGAN.

Generic Steganographic Generator

The generator architecture of our proposed StegaStyleGAN is outlined in Figure 3, which consists of the original SytleGAN2 generator G and an attached secret data modulating module (i.e., DP-SDM). Specifically, herein the G includes a mapping network f and a synthesis network g, where f consists of 8 fully connected layers and g consists of multiple stacked general blocks that progressively increase the resolution ($\uparrow 2^2\times$ by each block) of the generated images. The input of f is a 512-d latent code $\mathbf{z}$, which will be finally mapped into a 512-d intermediate vector $\mathbf{w}$. The input of g includes a learned $512 \times 4 \times 4$ constant tensor, a series of injected noise $N_i \sim \mathcal{N}(0,1)$ with different size, and the $\mathbf{w}$. The latent $\mathbf{z}$ determines the style of generated images, while the injected noise N, broadcasted to all feature maps of g by B and then added to the output of the corresponding convolution, is for generating stochastic detail. Moreover, the ablation experiments in StyleGAN further show that the finer (i.e., larger resolution) the injected noise, the better the control of image details (i.e., texture). On the other hand, the current cover-modified steganographic schemes, dedicated to embedding secret data into textured regions with rich redundant information, have shown excellent resistance to steganalyzers and good visual quality. Therefore, embedding data by controlling the generation of image details with the aid of injected noise N would be a promising approach.

Following this approach, the most straightforward way is to replace the injection noise with the secret data, but this hardly ensures the security of generative image steganography. As such, additional targeted design is essential (see GSN(Wei et al. 2022) for instance). Referring to the information-theoretic definition of steganographic security in (Cachin 2004), a stegosystem against passive adversaries is called $\varepsilon - secure$ if the KL-divergence between the probability distributions of cover (P_C) and stego (P_S) satisfies

$$D_{KL}(P_C, P_S) \leqslant \varepsilon, \tag{1}$$

and called perfectly secure if $\varepsilon = 0$. In our StegaStyleGAN, the cover and stego are generated by G($\mathbf{z}, \{N\}$) and G($\mathbf{z}, \{N\}, \mathbf{m}$), respectively, thus, to achieve secure steganography, making probability distribution $P_{(\{N\},\mathbf{m})} = P_{\{N\}}$ would be a feasible approach.

Along this point, as shown in Figure 3, without loss of generality, take the injected noise of k^{th} general block of G (i.e., N_k) for instance, we develop a **Distribution-Preserving Secret Data Modulator (DP-SDM)** attached to it. And following this way, the secret data $\mathbf{m}$ will and will only be integrated with the k^{th} injected noise in our model, i.e.,

$$N_k{}' = |N_k| \circ (2 * \mathbf{m} - \mathbf{1}), \tag{2}$$

where N_k is the k^{th} injected noise matrix, $\mathbf{m}$ is the binary secret data matrix with the same size of N_k, $N_k{}'$ is the modulated noise matrix, $\circ$ and $|\cdot|$ indicate the Hadamard product and absolute value operations, respectively. The larger the resolution of N_k, the larger the embedding capacity. The injected noise N in StyleGAN2 is always made to follow $\mathcal{N}(0,1)$ (standard normal distribution), the $\mathbf{m}$ in generative steganography is usually assumed to follow $\mathcal{B}(n, 0.5)$ (Bernoulli distribution). Under the modulation in Eq. (2), the distribution of $N_k{}'$ will be always identical to N_k. The proof is as follows:

Proof 1. Since $N_k \sim \mathcal{N}(0,1)$ *and* $\mathbf{m} \sim \mathcal{B}(n, 0.5)$, *the p.d.f. of* $N_k(i)$ *and* $\mathbf{m}(i)$ *have* $P_{N_k(i)=x} = P_{N_k(i)=-x}$ *and* $P_{\mathbf{m}(i)=0} = P_{\mathbf{m}(i)=1} = 0.5$, *respectively. Let* $\mathbf{T} = 2 * \mathbf{m} - 1$, *the p.d.f. of* $\mathbf{T}(i)$ *will have* $P_{\mathbf{T}(i)=-1} = P_{\mathbf{T}(i)=1} = 0.5$. *Then, for any* $i \in \{1, 2, ..., n\}$, *according Eq. (2), the p.d.f. of* $N'_k(i)$ *will have*

$$\begin{aligned} \because P_{N'_k(i)=x} &= P_{N_k(i)=|x|} * P_{\mathbf{T}(i)=-1} \\ &\quad + P_{N_k(i)=|x|} * P_{\mathbf{T}(i)=1} \\ &= P_{N_k(i)=|x|} * 0.5 + P_{N_k(i)=|x|} * 0.5 \\ &= P_{N_k(i)=|x|} \\ &\triangleq P_{N_k(i)=x} \quad (\Delta : P_{N_k(i)=x} = P_{N_k(i)=-x}) \\ \therefore P_{N'_k} &= P_{N_k} \end{aligned} \tag{3}$$

□

Since the distributions of generated covers and hidden content are the same, i.e., $D_{KL}(P_C, P_S) = 0$, then provably secure steganography is achieved. Note that for steganography with a larger embedding capacity of p bpp ($p \geqslant 2$, bits per pixel), the number of channels of $N_k{}'$ will change from 1 to p accordingly. For broadcasting it to all corresponding feature maps, we first divide these feature maps into multiple slices, each with p feature maps, and then traverse all the slices to add the p injected noise to the corresponding p feature maps. The visual schematic is shown in Figure 4.

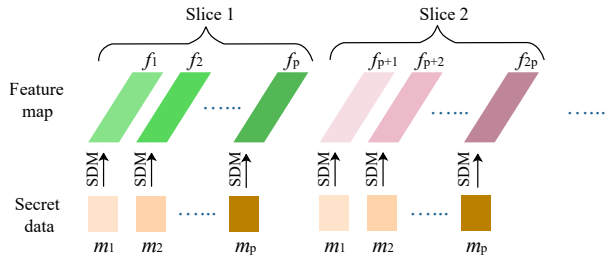

Figure 4: Schematic of secret data embedding.

Designs for Channel Attacks Resistance

Currently, most communication channels are lossy, especially the popular Online Social Network channels, where the transmitted images often suffer from some common channel attacks, such as JPEG compression, noise addition, rotation, filtering, scaling, etc. In this regard, we employ a tailored Image Attack Simulator (IAS) module to assist the training of G to generate robust stego images. Specifically, it consists of some common image attack layers, including JPEG compression, noise addition, rotation, filtering, and scaling. Since real JPEG compression is non-differential, in this IAS, we will refer to the pipeline proposed in (Zhang et al. 2021) to design the simulated JPEG compression layer.

As we know, in StyleGAN, the noise N injected to the general blocks with resolution 64^2 - 1024^2 and 4^2 - 32^2 are fine noise and coarse noise, respectively, they separately affect the medium-to-high and the medium-to-low frequency parts of generated images. And digital image watermarking studies have shown that watermarking in the medium-to-high frequency parts of an image can hardly resist some common image attacks, current image watermarking algorithms prefer to embed the watermarks in the medium-to-low frequency parts (Wang, Ni, and Huang 2012; Lu et al. 2022). On the other hand, the embedding of high-resolution layers is accompanied by a large embedding capacity, which will undoubtedly increase the difficulty of secret data extraction in the presence of channel attacks. As such, it is suggested to embed the secret data into the low-resolution layers with smaller embedding capacity so that the proposed model can more easily achieve robust steganography.

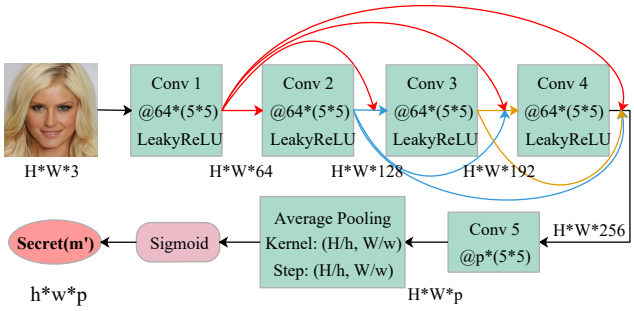

Figure 5: The architecture of our Secret Data Extractor.

Generic Secret Data Extractor

Figure 5 depicts the design of the Secret Data Extractor (SDE), whose backbone is a fully convolutional dense network (Huang et al. 2017) with $H \times W \times 3$ stego images as input and $h \times w \times p$ secret data matrix extracted by the network as output. SDE includes 5 convolutional layers with a kernel size of 5×5 and an average pooling layer, in which the first four convolutional layers are densely connected in the channel dimension to extract information, and the fifth convolutional layer achieves the final secret information output by compressing the channels. Dense connect designs aim to assist the network in predicting stronger forward information flow and train with better reverse gradient transfer, thereby improving the accuracy of data extraction. As for the final average pooling layer, it is specifically designed for robust steganography in the same framework, since the size of secret data in robust steganography under this scheme may be several equal divisions of the size of stego image (embedded in the non-last block of G). In addition, since the outputs of the pooling layer are floating-point numbers, the decoded $\mathbf{m}'$ requires additional processing during the evaluation phase, as follows:

$$\hat{\mathbf{m}} = Round(\mathbf{m}'). \tag{4}$$

In general, for this type of generative steganography approaches, including (Hu et al. 2018; Yu et al. 2021), (Wang et al. 2018), (Zhang et al. 2019b), (Wei et al. 2022), etc., all the extractors are unable to recover the original secret data with complete accuracy. Therefore, in practice, we also need to pre-process the secret data $\mathbf{m}$ with channel coding (e.g., RA, BCH, RS.) before embedding, and post-process the extracted $\hat{\mathbf{m}}$ with corresponding channel decoding.

Loss Functions and Training Strategy

The overall optimized loss function of our proposed StegaStyleGAN scheme mainly includes 3 components: 1) the loss of generator G: $\mathcal{L}_G$; 2) the loss of discriminator D: $\mathcal{L}_D$; and 3) the decoding loss of extractor E: $\mathcal{L}_E$ / $\mathcal{L}_{R_E}$. Their specific forms are as follows:

$$\begin{aligned}\mathcal{L}_G &= \mathcal{L}_{Adv_G} + \alpha * R_{PL} \\ &= \mathbb{E}_{z\sim p_z}\left[-log(D(G(\mathbf{z}, \mathbf{m})))\right] + \alpha * R_{PL},\end{aligned} \tag{5}$$

$$\begin{aligned}\mathcal{L}_D &= \mathcal{L}_{Adv_D} + \gamma * R_1 \\ &= \mathbb{E}_{z\sim p_z}\left[-log(1 - D(G(\mathbf{z}, \mathbf{m})))\right] \\ &+\mathbb{E}_{x\sim p_x}\left[-log(D(x)\right] + \gamma * R_1,\end{aligned} \tag{6}$$

$$\mathcal{L}_E = CrossEntropy(m, E(G(\mathbf{z}, \mathbf{m}))), \tag{7}$$

$$\mathcal{L}_{R_E} = CrossEntropy(m, E(Round(G(\mathbf{z}, \mathbf{m})))), \tag{8}$$

where $\mathcal{L}_{Adv_G}$ and $\mathcal{L}_{Adv_D}$ are the adversarial loss of G and D, respectively, R_{PL} is the path length regularization proposed in StyleGAN2 aiming to improve the training stability, R_1 is the regularization item developed in (Mescheder, Geiger, and Nowozin 2018) for accelerating the convergence of StyleGAN2. $\mathcal{L}_E$ and $\mathcal{L}_{R_E}$ are the binary cross entropy loss between the input $\mathbf{m}$ and the extracted $\mathbf{m}'$.

For the training of the proposed StegaStyleGAN, we divide each iteration into 4 steps, which are elaborated below:
Step 1: Freeze the G and E, and minimize $\mathcal{L}_D$, i.e.,

$$\min_D \mathcal{L}_D = \mathcal{L}_{Adv_D} + \gamma * R_1, \tag{9}$$

where the R_1 regularization participates in training every 16 iterations following the strategy of lazy regularization in StyleGAN2.

Step 2: Freeze the D and E, and minimize $\mathcal{L}_G$, i.e.,

$$\min_G \mathcal{L}_G = \mathcal{L}_{Adv_G} + \alpha * R_{PL}. \tag{10}$$

Similarly, the R_{PL} regularization in $\mathcal{L}_G$ participates in training every 4 iterations.

Step 3: Freeze the D, and minimize $\mathcal{L}_E$, i.e.,

$$\min_{G,E} \ \lambda * \mathcal{L}_E, \tag{11}$$

This step aims to not only optimize the E to improve its decoding accuracy but also further tune the parameters of the G to make it more favorable for the decoding of E.

Step 4: Freeze the G and D, and only fine-tune the E, i.e.,

$$\min_E \ \mathcal{L}_{R_E}. \tag{12}$$

This step is mainly to alleviate the effect of pixel rounding in image preservation on data extraction as the pixel values are integers. The rounding operation is non-differential, so leaving G behind and optimizing E alone is a last resort.

Experiments and Analysis

Datasets The entire CelebA (Liu et al. 2015) dataset will be resized and then cropped to 32^2 and 128^2 resolution to serve as training sets. In addition, 200k images sampled from the Lsun-bedroom (Yu et al. 2016) will be resized to 256^2 resolution to serve as training sets as well.

Implementation Details Our model is implemented in Pytorch and trained on 1 NVIDIA RTX 3090 GPU. The batch size is set to 16. The Adam optimizer with $\beta_1 = 0.0$, $\beta_2 = 0.99$ and $\varepsilon = 10^{-8}$ is used in training. The learning rates for the generator, discriminator, and extractor are all set as 0.0002. The hyper-parameters α and γ are set to 2 and 10, respectively. As for the λ, it should be initialized with a large value to ensure the $\mathcal{L}_E$ be competitive with $\mathcal{L}_{Adv_D}$ at the early training stage, then decay it once every 50 iterations, i.e., $\lambda = \lambda_{init} * 0.98^{\lfloor Iter/50 \rfloor}$, and stop if it less than a given lower bound. 250k iterations of StyleGAN2 pre-training will be performed prior to training the proposed StegaStyleGAN, with which to initialize the parameters of StegaStyleGAN.

Evaluation Metric Similar to the previous arts, we use Fréchet Inception Distance (FID), extraction accuracy of data (Acc), and the detection error rate of steganalyzer (P_e) to quantify the performance. In the proposed StegaStyleGAN, the FID is calculated using 50k training images in the dataset and 50k generated images, and the lower FID implies the generated images are more realistic. Acc is the ratio of the number of correctly extracted bits to the total number of embedded secret bits, and the higher the better. SCRMQ1 (Goljan, Fridrich, and Cogranne 2014), YeNet (Ye, Ni, and Yi 2017) and SRNet (Boroumand, Chen, and Fridrich 2019) steganalyzers are introduced to report P_e, and the closer P_e is to 0.5, the more secure the steganographic scheme is.

Dataset	Capacity (bpp)	GSN (Wei et al. 2022)			StegaStyleGAN-Ls		
		FID	Acc	CR_{ub}	FID	Acc	CR_{ub}
CelebA 128^2	1	13.29	97.53	0.833	**6.12**	**97.75**	**0.845**
	2	15.17	81.61	0.312	**6.37**	**97.41**	**0.844**
	4	16.21	70.14	0.120	**13.55**	**94.25**	**0.683**
Lsun 256^2	1	13.21	97.25	0.818	**5.59**	**98.63**	**0.896**
	2	14.56	83.19	0.347	**5.73**	**98.34**	**0.878**
	4	15.77	72.13	0.146	**10.31**	**95.65**	**0.742**

Table 1: Performance comparison of StegaStyleGAN-Ls and GSN on CelebA 128^2 and Lsun-bedroom 256^2.

Steganography under Lossless Channel

To evaluate the FID, P_e, and Acc of the proposed StegaStyleGAN-Ls model for lossless Channel covert communication, we train it separately on CelebA 128^2 and Lsun-bedroom 256^2 with embedding capacities of 1bpp, 2bpp, and 4bpp. Due to space constraints, only the comparison experiments with the SOTA model GSN are presented here (see table 1).

As shown in table 1, at 1bpp and 2bpp embedding capacity, the FID of our StegaStyleGAN-Ls is not only significantly lower than that of GSN but also very close to that of the original StyleGAN2 (CelebA: 5.1; Lsun: 3.7). This is because, instead of modifying the generator of the original StyleGAN2 by designing a secret block module equipped with a Hierarchical Gradient Decay skill as in GSN, the StegaStyleGAN-Ls fully inherits the network topology of StyleGAN2 generator without any modifications while keeping the distribution of the model inputs unchanged during steganographic embedding. And it is thanks to the introduction of DP-SDM to keep the distribution of model inputs unchanged, our model can be perfectly resistant to the detection of the most advanced SRNet (P_e=0.5)[1]. As for Acc, it decreases notably in GSN as the embedding capacity increases, while it stays above 94% in our StegaStyleGAN along with better FID than GSN. Moreover, the stego images generated by our StegaStyleGAN-Ls model are quite realistic-looking and can hardly be distinguished from the natural image by the naked eye.

Note that Acc only qualitatively reflects the steganographic ability, but not quantitatively. This is because, as mentioned before, the secret data $\mathbf{m}$ usually needs to be preprocessed by channel coding to ensure accurate information extraction, as such, the number of information bits k_m in $\mathbf{m}$ is the real steganographic capacity. When $\mathbf{m}$ of length L_m is given, the higher the *code rate* (k_m/L_m) of the channel coding, the larger k_m will be, as well as the steganographic capacity. To quantitatively compare the steganographic capacity of the two models, we further introduced the *code rate upper bound* (CR_{ub}) evaluation metric. Referring to the Shannon Bound, we know that the upper bound on the code

[1]For all steganalyzers, our P_e is close to 0.5. Moreover, for simplicity and space constraints, we can only report SRNet's results in the context here and cannot list them in table 1.

Model	Capacity (bpp)	FID	P_e	Acc										
				QF100	QF95	QF90	QF85	QF80	QF75	QF70	QF65	QF60	QF55	QF50
CIS-Net (You et al. 2022)	$3.1\text{x}10^{-2}$	6.20	1.0	97.98	97.92	97.81	97.70	97.52	97.28	97.01	96.72	96.38	95.94	95.46
StegaStyleGAN-Ly(32^2)	$6.25\text{x}10^{-2}$	3.74	0.5	99.75	99.79	99.75	99.76	99.73	99.71	99.62	99.58	99.49	99.44	99.36
StegaStyleGAN-Ly(32^2)	$1.25\text{x}10^{-1}$	4.78	0.5	99.63	99.62	99.63	99.59	99.54	99.49	99.42	99.31	99.14	98.91	98.64
StegaStyleGAN-Ly(128^2)	$6.25\text{x}10^{-2}$	9.58	0.5	99.47	99.46	99.55	99.47	99.28	98.87	–	–	–	–	–

Table 2: Performance comparison of the proposed StegaStyleGAN-Ly and CIS-Net against JPEG compression attack of different strengths.

Model	Capacity (bpp)	FID	P_e	Acc								
				Rotation	Gauss. Noise	Salt& Pepper	Speckle	Median	Mean	Gauss. Filter	Scale 0.5x	Scale 2.0x
Xue et al. (Xue and Wang 2021)	$6.71\text{x}10^{-4}$	–	–	78.18	91.57	92.20	92.20	92.36	91.84	92.16	91.55	93 <
Cao et al. (Cao et al. 2020)	$5.47\text{x}10^{-2}$	–	–	83.24	3.62	50.96	8.93	81.87	75.76	82.57	41.08	94 <
CIS-Net (You et al. 2022)	$1.56\text{x}10^{-2}$	6.94	1.0	98.18	96.74	98.93	98.92	98.15	98.12	99.19	–	–
StegaStyleGAN-Ly(32^2)	$6.25\text{x}10^{-2}$	3.74	0.5	99.61	97.11	99.35	96.74	98.91	99.63	99.64	98.83	99.71
StegaStyleGAN-Ly(128^2)	$6.25\text{x}10^{-2}$	9.58	0.5	99.59	90.02	99.61	91.99	98.57	99.59	99.62	99.03	99.68

Table 3: Performance comparison of the proposed StegaStyleGAN-Ly and other competed schemes against other noise attacks.

rate for optimal channel coding is

$$CR_{ub} = 1 - H_2(Acc), \tag{13}$$

where $H_2(p) = -p * \log_2 p - (1-p) * \log_2(1-p)$. According to the performance of CR_{ub} in table 1, the steganographic capacity of the proposed StegaStyleGAN will be much higher than that of GSN, which can theoretically reach 2.5+ times and 5+ times that of GSN at 2bpp and 4bpp embedding capacities, respectively.

Steganography under Lossy Channel

To evaluate the FID, P_e, and Acc performance of the proposed StegaStyleGAN-Ly model for lossy Channel, we train it on CelebA 32^2 and CelebA 128^2, respectively. The previous art CIS-Net (You et al. 2022), whose generated images are of only 32^2 resolution, is first introduced for comparison. To deal with the common JPEG compression attack, unlike in CIS-Net where several QF candidates are chosen to simulate JPEG compression, we train with only one, i.e., $QF = 50$ for StegaStyleGAN-Ly(32^2) and $QF = 75$ for StegaStyleGAN-Ly(128^2). To further enhance its robustness against other noise attacks, we follow the two-stage training strategy proposed in (Liu et al. 2019) as well. Specifically, we train the StegaStyleGAN-Ly(32^2) and StegaStyleGAN-Ly(128^2) only with JPEG compression attack in the first stage, and then freeze the generator and fine-tune the extractor for the other noise attacks separately in the second stage. The embedding layer of StegaStyleGAN-Ly(32^2) and StegaStyleGAN-Ly(128^2) are 8^2 and 32^2 resolution layers, respectively. The experimental results are collected in table 2 and table 3. Note that, all the schemes involved in the comparison use the same image attack strength settings in both training and testing (see (Xue and Wang 2021; Cao et al. 2020) for details).

As shown in table 2, compared with CIS-Net, the stego images generated by StegaStyleGAN-Ly(32^2) have better visual quality despite embedding more secret data and are perfectly resistant to detection by SOTA steganalyzers. More importantly, for all strengths of JPEG compression attacks listed in table 2, the StegaStyleGAN-Ly(32^2) still shows higher Acc than CIS-Net even though it only uses one QF for training. In addition, the large-resolution StegaStyleGAN-Ly(128^2) still has excellent performance of FID, P_e, and Acc with an embedding capacity of 1024 bits, indicating the proposed StegaStyleGAN-Ly has higher practical potential than CIS-Net. As for the cases of against noise attacks other than JPEG compression, we refer to the same settings in (Xue and Wang 2021; Cao et al. 2020; You et al. 2022), and compare our StegaStyleGAN-Ly with them, where the noise attacks consist of image rotation (50°), the addition of various noises, including Gaussian ($\sigma = 0.1$), Salt & Pepper ($p = 0.005$), and Speckle ($\sigma = 0.1$), various kinds of filtering (size 3×3), including Median, Mean, and Gaussian, and scaling with factor 0.5 and 2. Referring to the results in table 3, we can find that our proposed StegaStyleGAN-Ly(32^2) can not only significantly outperform (Xue and Wang 2021) and (Cao et al. 2020), but also still compete with CIS-Net in terms of Acc performance against the involved attacks despite embedding larger capacity while maintaining a significant advantage in terms of FID and P_e performance. In addition, the large-resolution StegaStyleGAN-Ly(128^2) further shows excellent practical potential in resisting other noise attacks.

Dataset	DP-SDM	Capacity	FID	P_e	Acc
CelebA 128^2	w/	1 bpp	6.12	**0.501**	97.75
	w/o	1 bpp	6.15	1.0	98.01

Table 4: Performance of StegaStyleGAN-Ls trained with (w/) and without (w/o) DP-SDM.

Dataset	Fine-tuning E	1 bpp	2 bpp	4 bpp
CelebA 128^2	w/	97.75	97.41	94.25
	w/o	96.27	95.96	93.26

Table 5: Performance of StegaStyleGAN-Ls trained with (w/) and without (w/o) the fine-tuning of E.

Ablation Study

Effect of Distribution-Preserving Secret Data Modulator Take the model trained on CelebA 128^2 with 1bpp embedding capacity for experiments. The performance of StegaStyleGAN-Ls with and without DP-SDM are collected in table 4. It is observed that although the DP-SDM can not advantage FID and Acc, it does make the generative steganography secure.

Effect of Fine-tuning of E Take the StegaStyleGAN-Ls model trained on CelebA 128^2 with 1bpp, 2bpp and 4bpp embedding capacity for experiments. Its performance trained without the fine-tuning of E is collected in table 5. Note that all hyperparameters have not been re-optimized. Referring to the results in table 5, it is observed that the fine-tuning of E can indeed improve its Acc.

Influence of Secret Data Distribution

The above *proof 1* states that the provably secure generative steganography is guaranteed by the strict assumption that the secret data $\mathbf{m}$ follows $\mathcal{B}(n, p = 0.5)$. In practice, however, the $\mathbf{m}$ distribution is often slightly biased, i.e., $p \neq 0.5$. Therefore, investigating the influence of $\mathbf{m}$ distribution on the performance of the StegaStyleGAN scheme is vitally important. To this end, we train the StegaStyleGAN-Ls on CelebA 128^2 and Lsun-bedroom 256^2 with $\mathbf{m} \sim \mathcal{B}(n, p = 0.5)$ at embedding capacity of 1bpp, 2bpp, and 4bpp, and then test it separately with $\mathbf{m} \sim \mathcal{B}(n, p = 0.45)$ and $\mathbf{m} \sim \mathcal{B}(n, p = 0.4)$. The experimental results are collected in table 6. Comparing the results in table 1 and table 6, it can be seen that the proposed StegaStyleGAN maintains its original performance even if the distribution of input $\mathbf{m}$ at testing has some deviations compared to that at training.

Generative Steganography Meets GAN Inversion

Focusing on enabling the generated stego image to have realistic semantics (i.e., conditional generation), thereby promoting generative steganography towards practicality, we present a promising scheme by mating with the GAN Inversion. In specific, given a real target image $\mathbf{x}$, we seek to find the corresponding $\mathbf{w} \in \mathcal{W}$ and the per-layer injected noise N_i in the StegaStyleGAN to synthesize a stego image with realistic semantics that closely resembles $\mathbf{x}$. Note that previous research (Abdal, Qin, and Wonka 2019; Gabbay and Hoshen 2019) suggests that the separate $\mathbf{w}$ for each layer of the generator can improve the inversion results. The N_i and $\mathbf{w}$ are initialized by the standard normal distribution and the off-the-shelf $u_{\mathbf{w}}$, where $u_{\mathbf{w}} = \mathbb{E}_{\mathbf{z}} f(\mathbf{z})$ is pre-computed by randomly running 10k latent code $\mathbf{z}$ through the f. The loss function for StegaStyleGAN Inversion consists of two parts: 1) image quality term (LPIPS(Zhang et al. 2018)), 2) injected noise regularization term, which will be optimized by the gradient-based method (Karras et al. 2020). To accomplish the Inversion of StegaStyleGAN, we will freeze the parameters of its generator and optimize only the corresponding $\mathbf{w}$ and N_i. The experiment results show that with the increase of training iterations of StegaStyleGAN inversion, the generated stego images gradually resemble the real target images along with promising Acc performance.

Dataset	Capacity	$p = 0.45$			$p = 0.4$		
		FID	P_e	Acc	FID	P_e	Acc
CelebA 128^2	1 bpp	6.10	0.5	97.43	6.05	0.5	97.26
	2 bpp	6.21	0.5	97.28	6.14	0.5	97.02
	4 bpp	13.40	0.5	93.33	13.05	0.5	92.62
Lsun 256^2	1 bpp	5.71	0.5	98.41	5.72	0.5	98.30
	2 bpp	5.81	0.5	98.32	5.79	0.5	98.13
	4 bpp	10.01	0.5	95.27	10.03	0.5	94.78

Table 6: Performance of StegaStyleGAN-Ls trained and tested with mismatched parameters $\mathbf{m}$.

Conclusion

Towards generic and practical generative image steganography, we propose a novel scheme, namely StegaStyleGAN, in this paper, which can meet the practical objectives of security, capacity, and robustness within the same framework. In our StegaStyleGAN, secret data is used to modulate the injected noise of the StegaStyleGAN with a Distribution-Preserving Secret Data Modulator (DP-SDM), rather than directly as input to the generator, thus enabling visually convincing and provably secure generative image steganography. To accurately extract the embedded secret data, a generic and efficient Secret Data Extractor (SDE) is also invented. In addition, a customized image attack simulator is employed in StegaStyleGAN to aid in its training, showing excellent effects in improving its robustness to common image attacks. Moreover, by mating with GAN inversion, the StegaStyleGAN can generate stego images with realistic semantics, further highlighting its practical potential. For lossless covert communication, our StegaStyleGAN-Ls model can generate higher-quality stego images with larger steganographic capacities compared to the SOTA GSN. And for lossy covert communication, the proposed StegaStyleGAN-Ly model is more practical than the SOTA CIS-Net with larger steganographic capacities as well. More supplementary material can be found at https://github.com/vazswk/StegaStyleGAN.git.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (Grants No. 62202507, U22A2030, U1936212, and U23B2022), Natural Science Foundation of Guangdong Province, China (Grant No. 2022A1515011209), and China Postdoctoral Science Foundation (Grant No. 2021M703767).

References

Abdal, R.; Qin, Y.; and Wonka, P. 2019. Image2StyleGAN: How to Embed Images Into the StyleGAN Latent Space? In *Proceeding of 2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019*, 4431–4440. Seoul, Korea (South): IEEE.

Baluja, S. 2020. Hiding Images within Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(7): 1685 – 1697.

Boroumand, M.; Chen, M.; and Fridrich, J. 2019. Deep residual network for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 14(5): 1181–1193.

Cachin, C. 2004. An information-theoretic model for steganography. *Inf. Comput.*, 192(1): 41–56.

Cao, Y.; Zhou, Z.; Wu, Q. M. J.; Yuan, C.; and Sun, X. 2020. Coverless information hiding based on the generation of anime characters. *EURASIP J. Image Video Process.*, 2020(1): 36.

Dong, L.; Wang, J.; Wang, R.; Li, Y.; and Sun, W. 2021. Towards Image Data Hiding via Facial Stego Synthesis with Generative Model. In *International Joint Conference on Artificial Intelligence - International Workshop on Safety & Security of Deep Learning, IJCAI Workshop 2021*. virtual: ijcai.org.

Filler, T.; and Fridrich, J. 2009. Fisher information determines capacity of ϵ-secure steganography. In *Proceedings of the 11th International Workshop on Information Hiding, IH 2009, LNCS, vol 5806*, 31–47. Darmstadt, Germany: Springer.

Fridrich, J. 1999. Applications of data hiding in digital images. In *Tutorial for the Fifth International Symposium on Signal Processing and its Applications, ISSPA 1999*, 22–25. Brisbane, QL, Australia: IEEE.

Fridrich, J., ed. 2009. *Steganography in Digital Media: Principles, Algorithms, and Applications*. Cambridgeshire, United Kingdom: Cambridge University Press.

Fridrich, J.; and Kodovský, J. 2012. Rich models for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 7(3): 868–882.

Gabbay, A.; and Hoshen, Y. 2019. Style Generator Inversion for Image Enhancement and Animation. arXiv:1906.11880.

Goljan, M.; Fridrich, J. J.; and Cogranne, R. 2014. Rich model for Steganalysis of color images. In *2014 IEEE International Workshop Information Forensics and Security, WIFS 2014*, 185–190. Atlanta, GA, USA: IEEE.

Holub, V.; Fridrich, J.; and Denemark, T. 2014. Universal distortion function for steganography in an arbitrary domain. *EURASIP Journal on Information Security*, 2014(1): 1 – 13.

Hu, D.; Wang, L.; Jiang, W.; Zheng, S.; and Li, B. 2018. A Novel Image Steganography Method via Deep Convolutional Generative Adversarial Networks. *IEEE Access*, 6: 38303–38314.

Huang, G.; Liu, Z.; van der Maaten, L.; and Weinberger, K. Q. 2017. Densely Connected Convolutional Networks. In *Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2261–2269. Honolulu, HI, USA: IEEE Computer Society.

Jing, J.; Deng, X.; Xu, M.; Wang, J.; and Guan, Z. 2021. HiNet: Deep Image Hiding by Invertible Network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV 2021*, 4713–4722. Montreal, QC, Canada: IEEE.

Karras, T.; Laine, S.; and Aila, T. 2019. A Style-Based Generator Architecture for Generative Adversarial Networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019*, 4401–4410. Long Beach, CA, USA: CVF/IEEE.

Karras, T.; Laine, S.; Aittala, M.; Hellsten, J.; Lehtinen, J.; and Aila, T. 2020. Analyzing and Improving the Image Quality of StyleGAN. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020*, 8107–8116. Seattle, WA, USA: CVF/IEEE.

Li, B.; Wang, M.; Huang, J.; and Li, X. 2014. A new cost function for spatial image steganography. In *Proceedings of the IEEE International Conference on Image Processing, ICIP 2014*, 4206 – 4210. Paris, France: IEEE.

Li, Z.; Zhang, M.; and Liu, J. 2021. Robust image steganography framework based on generative adversarial network. *J. Electronic Imaging*, 30(2): 023006.

Liu, Y.; Guo, M.; Zhang, J.; Zhu, Y.; and Xie, X. 2019. A Novel Two-stage Separable Deep Learning Framework for Practical Blind Watermarking. In *Proceedings of the 27th ACM International Conference on Multimedia, MM 2019*, 1509–1517. Nice, France: ACM.

Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2015. Deep Learning Face Attributes in the Wild. In *Proceedings of 2015 IEEE International Conference on Computer Vision, ICCV 2015*, 3730–3738. Santiago, Chile: IEEE Computer Society.

Lu, J.; Ni, J.; Su, W.; and Xie, H. 2022. Wavelet-Based CNN for Robust and High-Capacity Image Watermarking. In *Proceedings of IEEE International Conference on Multimedia and Expo, ICME 2022*, 1–6. Taipei, Taiwan: IEEE.

Lu, S.; Wang, R.; Zhong, T.; and Rosin, P. L. 2021. Large-Capacity Image Steganography Based on Invertible Neural Networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021*, 10816–10825. Virtual: CVF / IEEE.

Mescheder, L. M.; Geiger, A.; and Nowozin, S. 2018. Which Training Methods for GANs do actually Converge? In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, 3478–3487. Stockholmsmässan, Stockholm, Sweden: PMLR.

Su, W.; Ni, J.; Hu, X.; and Fridrich, J. 2020. Image Steganography with Symmetric Embedding using Gaussian Markov Random Field Model. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(3): 1001–1015.

Su, W.; Ni, J.; Hu, X.; and Huang, J. 2022. New design paradigm of distortion cost function for efficient JPEG steganography. *Signal Processing*, 190: 108319.

Su, W.; Ni, J.; Li, X.; and Shi, Y. Q. 2018. A New Distortion Function Design for JPEG Steganography using the Generalized Uniform Embedding Strategy. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(12): 3545 – 3549.

Tang, W.; Li, B.; Barni, M.; Li, J.; and Huang, J. 2021. An Automatic Cost Learning Framework for Image Steganography Using Deep Reinforcement Learning. *IEEE Transactions on Information Forensics and Security*, 16: 952–967.

Wang, C.; Ni, J.; and Huang, J. 2012. An Informed Watermarking Scheme Using Hidden Markov Model in the Wavelet Domain. *IEEE Transactions on Information Forensics and Security*, 7(3): 853–867.

Wang, Z.; Gao, N.; Wang, X.; Qu, X.; and Li, L. 2018. SSteGAN: Self-learning Steganography Based on Generative Adversarial Networks. In *Proceedings of the 25th International Conference on Neural Information Processing, ICONIP 2018*, 253–264. Siem Reap, Cambodia: Springer.

Wei, P.; Li, S.; Zhang, X.; Luo, G.; Qian, Z.; and Zhou, Q. 2022. Generative Steganography Network. In *Proceedings of 30th ACM International Conference on Multimedia, MM 2022*, 1621–1629. Lisboa, Portugal: ACM.

Xue, R.; and Wang, Y. 2021. Message Drives Image: A Coverless Image Steganography Framework Using Multi-Domain Image Translation. In *Proceedings of International Joint Conference on Neural Networks, IJCNN 2021*, 1–9. Shenzhen, China: IEEE.

Yang, J.; Ruan, D.; Huang, J.; Kang, X.; and Shi, Y. 2020. An Embedding Cost Learning Framework Using GAN. *IEEE Transactions on Information Forensics and Security*, 15: 839–851.

Ye, J.; Ni, J.; and Yi, Y. 2017. Deep learning hierarchical representations for image steganalysis. *IEEE Transactions on Information Forensics and Security*, 12(11): 2545–2557.

Ying, Q.; Zhou, H.; Zeng, X.; Xu, H.; Qian, Z.; and Zhang, X. 2021. Hiding Images into Images with Real-world Robustness. arXiv:2110.05689.

You, Z.; Ying, Q.; Li, S.; Qian, Z.; and Zhang, X. 2022. Image Generation Network for Covert Transmission in Online Social Network. In *Proceedings of the 30th ACM International Conference on Multimedia, MM 2022*, 2834–2842. Lisboa, Portugal: ACM.

Yu, C.; Hu, D.; Zheng, S.; Jiang, W.; Li, M.; and Zhao, Z. 2021. An improved steganography without embedding based on attention GAN. *Peer-to-Peer Netw. Appl.*, 14(3): 1446–1457.

Yu, F.; Seff, A.; Zhang, Y.; Song, S.; Funkhouser, T.; and Xiao, J. 2016. LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop. arXiv:1506.03365.

Zhang, C.; Benz, P.; Karjauv, A.; Sun, G.; and Kweon, I. S. 2020. UDH: Universal Deep Hiding for Steganography, Watermarking, and Light Field Messaging. In *Proceedings of the Advances in Neural Information Processing Systems 33: NeurIPS 2020*. Virtual: PMLR.

Zhang, C.; Karjauv, A.; Benz, P.; and Kweon, I. S. 2021. Towards Robust Deep Hiding Under Non-Differentiable Distortions for Practical Blind Watermarking. In *Proceedings of the 29th ACM International Conference on Multimedia, MM 2021*, 5158–5166. Virtual: ACM.

Zhang, K. A.; Cuesta-Infante, A.; Xu, L.; and Veeramachaneni, K. 2019a. SteganoGAN: High Capacity Image Steganography with GANs. arXiv:1901.03892.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *Proceedings of 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018*, 586–595. Salt Lake City, UT, USA: CVF/IEEE Computer Society.

Zhang, Z.; Liu, J.; Ke, Y.; Lei, Y.; Li, J.; Zhang, M.; and Yang, X. 2019b. Generative Steganography by Sampling. *IEEE Access*, 7: 118586–118597.