Stable NICE Model-Based Picture Generation for Generative Steganography

Xutong Cui[1], Zhili Zhou[2(✉)], Jianhua Yang[3], Chengsheng Yuan[1], and Weixuan Tang[2]

[1] Engineering Research Center of Digital Forensics, Ministry of Education, Nanjing University of Information Science and Technology, Nanjing 210044, China
yuancs@nuist.edu.cn

[2] Institute of Artificial Intelligence, Guangzhou University, Guangdong 510006, China
zhou_zhili@163.com, twei@gzhu.edu.cn

[3] School of Cyber Security, Guangdong Polytechnic Normal University, Guangdong 510006, China
yangjh86@gpnu.edu.cn

Abstract. Steganography is one of most important techniques for covert communication. In recent years, generative steganography, which transforms a secret information into a generated picture, is a prospective steganography-resistant technique. Nevertheless, it is difficult to achieve a good trade-off between information hiding ability and extraction accuracy because of the low efficiency and irreversibility of the secret-to-picture conversion. In order to solve this problem, this paper proposes a secret message-driven picture generation solution for generative steganography. The presented SM-IG scheme is founded on the design of a stable version of the Nearly Independent Component Estimation (Stable NICE) model, allowing for a stable bijection mapping between a potential space with simple distributions and an picture space with complex distributions. During the secret to picture conversion, a latent vector is constructed, driven by a given secret message, which is then mapped to the generated picture via the Stable NICE modelAs a result, the secret information is eventually converted into the generated picture. Due to the good efficiency and reversibility of the SM-IG scheme, this steganography method has high hiding capability and accurate message extraction accuracy. The experiments prove that the proposed SM-IG can simultaneously realise good-level hiding capacity (as much as 4 bpp) and precise extraction accuracy (close to 100% accuracy) without compromising the required resistance to detection and imperceptibility.

Keywords: Steganography · Generative steganography · Information hiding · Digital forensics

J. Cai et al. (Eds.): SPNCE 2023, LNICST 525, pp. 297–314, 2025.
https://doi.org/10.1007/978-3-031-73699-5_21

1 Introduction

Picture steganography is a technique for hiding secret messages in cover pictures, allowing secret communication without suspicion [1–3]. The advantage of steganography is that it hides the happening of secret communication and thus ensures the safety of secret message transmission [1–3].

Generally, the traditional steganography methods take an existing picture as a cover and secret message is embedded in it by modifying the cover picture. However, they will inevitably cause the cover picture to become distorted, especially with high concealed payloads. In this case, a well-designed implicit analyzer is able to detect the existence of hidden information [4]. Some researchers have proposed "generative steganography" to resist the detection of steganalyzers [5–7]. However, these methods can only convert short secret messages into low-dimensional inputs to the generative model, and can only construct one-way mappings between the input message and the picture content, making the secret to picture conversion process low efficiency and non-reversible. As a result, very limited hiding power or the inability to accurately extract secret messages from the generated picture limits the applicability of these methods to practical steganography tasks.

Based on the above analyses, in order to realize large-volume hiding and precise extraction of secret messages, how to design efficient reversible conversion between secret information and pictures is crucial for generating steganography. To this end, we design a flow-based generative model, *i.e.*., Stable version of Near Independent Component Estimation (Stable NICE) model, to enable a stable bijective mapping between high-dimensional potential vectors and pictures. Based on the designed Stable NICE model, we propose the SM-IG scheme to achieve an efficient and reversible conversion between the secret information and produced pictures for messages hiding and extraction.

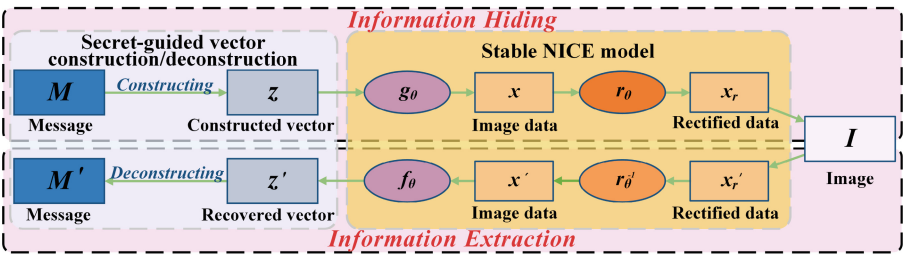

Fig. 1. The framework of SM-IG scheme for generative steganography based on the Stable NICE model.

Figure 1 illustrates the structure of the SM-IG steganography generation scheme on the basis of the stable NICE model. The proposed steganographic approach has the following advantages over existing generative steganographic approaches.

(1) The proposed SM-IG scheme demonstrates high efficiency and reversibility, resulting in a maximum high hiding capacity (up to 4 *bpp*) and almost 100% accuracy in the extraction of secret messages.
(2) The pictures produced can remain high quality as the hidden payload grows.
(3) Existing steganographers are still unlikely to defeat the proposed steganography method because the steganography method generates a new picture as a steganographic picture instead of modifying the existing picture for information hiding.

2 Related Work

Generative steganography, different from conventional steganography, allows you to hide secret messages without having to change the cover picture, providing a promising performance of anti-detectability. Existing methods for generative steganography can be broadly classified into two categories: methods based on texture synthesis and methods based on GANs.

Xu *et al.* [8] converted secret messages directly into complex texture pictures for steganography. Li *et al.* [9] converted the secret information to a fingerprint picture by mapping it to fingerprint phase. However, the meaningless of texture pictures could arouse the suspicion of attackers.

Luckily, the modern generative deep learning models, *i.e.*, Generative Adversarial Networks (GANs) [10], are capable of producing new significant pictures which look very real, *i.e.*, "realistic-looking pictures". Consequently, a number of generative steganographic approaches converted the secret messages into the realistic-looking pictures based on GANs. Cao *et al.* [11] converted a given secret message to attribute labels of the anime characters and generated anime characters with GAN [12] constrained by the labels. Then, Illustration2Vec [13] is used to extract the labels of anime characters and then convert them to secret messages by long short-term memory network (LSTM).Qin *et al.* [14] established a mapping dictionary between object labels and binary message sequences after detection by using Faster RCNN [15], and then employed the GANs to generate stego-picture with corresponding labels. However, the hiding capacity of these generative steganography methods is very limited, since very limited amount of information can be carried by simple labels and semantic information.

To enhance hiding capacity, some picture generative steganographic approaches constructed a mapping between secret message and a noise signal, and then fed the noise signal to GANs for stego-picture generation. Hu *et al.* [7,16] fed the noise signal encoded by the secret message to Deep Convolutional GAN (DCGAN) [17], since the DCGAN has the ability of generating high-quality stego-pictures that looks realistic. To improve the quality of the generated picture, Li *et al.* [18] adopted Wasserstein GAN Gradient Penalty (WGAN-GP) [19] instead of DCGAN for steganography. Arifianto *et al.* [20] converted the secret message to a word vector by employing a word2vec model [21]. The word vector is fed to the GANs to generate the stego-picture. However, GANs only establish a unidirectional mapping relationship from low-dimensional input information

to high-dimensional pictures. Thus, the above GANs-based steganography methods may fail to precisely recover secret messages from these produced pictures, particularly when the payload is high.

In conclusion, it is difficult for the current generative steganographic methods to realise a feasible balance between hiding power and secret message extraction accuracy, which restricts the task of steganography in practice.

To achieve accurate extraction and high-capacity hiding of secret information simultaneously, in this paper, we attempt to design the Stable NICE model to enable a stable bijective-mapping between high-dimensional latent vectors and pictures. Then, based on the designed Stable NICE model, we propose the SM-IG scheme to realize an efficient and reversible transformation between the secret message and generated picture for information hiding and extraction. Consequently, while maintaining the anti-detectability and imperceptibility desirable for generative steganography, the suggested method achieves high-level information hiding capacity and precise message extraction simultaneously.

3 Designed Stable Nice Model

In this paper, the SM-IG scheme is proposed for picture generative steganography, which is implemented by a flow-based model. Thus, to implement the scheme, we first design a proper flow model, *i.e.*, Stable NICE model, to realize a stable bijective-mapping between latent space and picture space.

NICE model enables a bijective-mapping between picture space and latent space. However, as the bijective-mapping is not stable enough, directly applying the original NICE model for the SM-IG scheme will affect the information extraction significantly. Further details are below.

As Fig. 1 illustrates, in the secret message to picture conversion process of SM-IG scheme, driven by a provided secret message, a latent vector z is constructed and then mapped to an picture data x for picture generation. The generated picture is then reverse mapped to the latent vector to extract information. If the original NICE model is employed to map the constructed latent vector z to the picture data x, it is found that x contains some exceptional elements, which are out of the value range [0,1]. That is mainly caused by the finite size of training dataset and the errors of bijective-mapping between continuous latent space and discrete picture space. To generate the picture from x, the original NICE model directly removes these exceptional elements in x. The information loss of x makes the recovered latent vector $z^{'}$ quite different from the original one z, as shown in Fig. 2(a). That will affect the extraction of secret message significantly. Therefore, it is not a good choice to directly apply the original NICE model to implement the SM-IG scheme for generative steganography.

To address this issue, it is straightforward to normalize all the values of picture data x to the range of [0,1] for picture generation. However, if some exceptional elements of x have very large or small values, the normalization causes a lot of elements of x to be changed significantly, which will decrease the quality of generated picture, as shown in Fig. 2. To accurately extract secret message

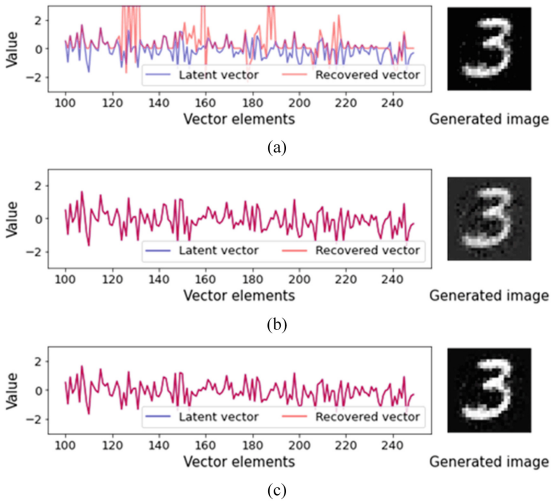

Fig. 2. Different recovered latent vectors and generated pictures, which are obtained by (a) NICE model with removal of exceptional elements of picture data, (b) NICE model with picture data normalization, and (c) NICE model with picture data rectification (Stable NICE model), respectively.

without affecting the qualities of generated pictures, we design a stable version of NICE model, called as Stable NICE model, by adding a pair of rectification functions into the original NICE model, as shown in Fig. 1. The rectification functions are described as follows.

Instead of simply normalizing the original picture data x, an invertible rectification function r_α is learned to rectify x before picture generation, and its inverse function is used to recover the original picture data x from the generated picture. The rectification function and its inverse function are defined by

$$x_r = r_\Theta(x) = [\tanh(\alpha_1 x + \alpha_2) + 1]/2 \tag{1}$$

$$x = r_\Theta^{-1}(x_r) = \frac{1}{2\alpha_1} \log_2 \left(\frac{x_r}{1 - x_r} \right) - \frac{\alpha_2}{\alpha_1} \tag{2}$$

The function $r_\Theta(\cdot)$ensures that the exceptional elements of x are rectified to the range of [0,1] while most of other elements are adjusted slightly so as to generate the high-quality picture; Its inverse function $r_\Theta^{-1}(\cdot)$ allows accurate recovery of original picture data and latent vector for information extraction.

To learn the function parameters, we consider two objectives: (1) Minimizing the difference between the recovered latent vector $z^{'}$ and the original one z, and (2) Minimizing the difference between the picture I generated by rectifying the data x and the picture I_0 generated by directly removing the exceptional elements of x. Thus, the function parameters are learned by Genetic algorithm using the following equation. More details of learning process are given in Sect. 5.

$$\min_{\alpha \in A}(\|z^{'} - z\| + \|I - I_0\|) \tag{3}$$

Owing to the bijective-mapping functions of original NICE model and the added rectification functions, the Stable NICE model can ensure good stability of bijective-mapping between latent space and picture space while maintaining high qualities of generated pictures, as shown in Fig. 2(c).

4 SM-IG Scheme for Generative Steganography

Figure 1 illustrates the structure of the SM-IG generative steganography scheme on the basis of the stable NICE model, which is composed of two phases: the secret to picture transform for message hiding and the secret to picture inverse transform for message extraction.

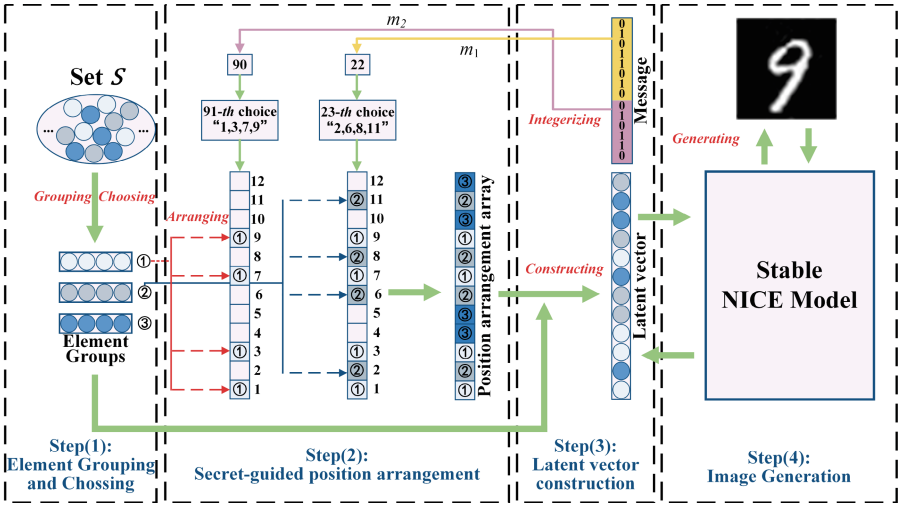

Fig. 3. The illustration of S2I transformation for information hiding.

4.1 S2I Transformation for Information Hiding

As shown in Fig. 3, the task of secret messages to pictures convert is to hide messages. First, we group a set of randomly sampled elements and choose a part of proper elements from each group. Then, a high-dimensional latent vector z is constructed by arranging the selected elements at the corresponding relative locations, driven by the given information. Finally, the built vector z is mapped to a generated picture I via the Stable NICE model. Consequently, long secret messages can be naturally hidden in the generated picture I.

Thus, the processing of the S2I transform consists of four steps: element grouping and selection, secret-guided location arrangement of elements, latent vector construction, and picture generation, as shown in Fig. 3. The specific steps are as follows.

Step (1): Element grouping and choosing. In NICE model, It is common to randomly sample a suite of N_T elements from the standard Gaussian distribution to build a N_T-dimensional potential vector for picture generation, as each component of latent space follows the standard Guassian distribution. Consequently, the values of randomly sampled elements follow an approximate Guassian distribution.

To hide a given secret message M, if we encode the message straight away as the elements for latent vector construction, the distribution of these elements is likely to be very different from the distribution of randomly sampled elements, *i.e.*, approximate Guassian distribution, which will pose threats to security of hidden message. To avoid this issue, do not directly encode secret messages as elements, we encode it as the corresponding location arrangement for the N_T randomly sampled elements to construct the latent vector z. More details are given as follows.

For a standard Guassian distribution, we first split the range of [-2,2] into K parts and make the probability values of each part to be equal, because the N_T elements sampled from the standard Gaussian distribution are mostly fall into the range. According to the K range parts, the N_T sampled elements are split into K groups, denoted as $\{G_i|1 \leq i \leq K\}$.

As shown in Fig. 2(c), using the stable NICE model for information extraction, while it is possible to recover the constructed potential vectors, they remain somewhat dissimilar to the recovered potential vectors. This is mainly because certain components of the created concealed vectors that are near group boundaries are displaced towards adjoining groups. This displacement is caused by the bijective mapping error between the concealed continuous space and the discrete pictures space. That will affect the accuracy of information extraction. Thus, select the n elements closest to the centre of each group to get N elements to construct latent vector, which are unlikely to be transferred to adjacent groups. Consequently, $N = K \times n$ and $N \leq N_T$. It is notable that K and n are shared between communication participants in advance to hide and extract information.

Step (2): Secret-guided location arrangement of elements. A set of corresponding locations, controlled by a given secret message M, is assigned to each group G_i of n elements. Select n locations in the location set *pos*, which registers the $[N - (i-1)n]$ locations remaining after ordering the elements, where $(i-1)n$ denotes the number of locations taken by the elements of $1-th$ to $(i-1)-th$ groups. The count of options to select n locations from *pos* can be calculated by $\omega_i = C(N-(i-1)n, n)$, where $C(x, y)$ denotes the count of options to select y locations from x locations. Therefore, $\lfloor \log_2 \omega_i \rfloor$-bit secret information can be encoded as the location order of elements of G_i.

The pseudo-code of Algorithm 1 details the secret bootstrap location arrangement for each group of elements. Also shown in Fig. 3 is an instance of a secret guide location arrangement. Assuming that $N = 12$ elements are selected from $K = 3$ groups $\{G_i|1 \leq i \leq 3\}$, with each group containing $n = 4$ elements. Since there are 12 elements to arrange, the original set of locations for the arrange-

Algorithm 1. Secret-guided location arrangement of elements

Input: Secret bitstream: $M =' 01011010010110'$, Number of groups: K, Number of chosen elements in each group: n, Number of chosen elements in total: N;
Output: Array of location arrangement: Ind=(Ind[1],Ind[2],...,Ind[N];

$r \leftarrow M$; ▷ Record number of remaining locations
$pos \leftarrow$ (1,2,3,...,N); ▷ Record remaining locations
for $i = 1$ to $K - 1$ **do**
  $\omega_i = C(r, n)$; ▷ Compute number of choices of selecting n locations from r locations
  $m \leftarrow$ Read($M, \lfloor \log_2 \omega_i \rfloor$); ▷ Read the next $\lfloor \log_2 \omega_i \rfloor$ bits from M as a positive decimal integer m
  $sel \leftarrow$ Select(m, pos, n); ▷ Select n locations from pos guided by m
  for p in sel **do**
    $Ind[p] \leftarrow i$; ▷ Set $Ind[p]$ as the group *No.*, *i.e.*, i
  end for
  $r \leftarrow r - n$;
  $pos \leftarrow$ PosDelete(pos, sel); ▷ Remove sel from pos
end for
return Ind

ment of these elements is represented as $pos = (1, 2, 3, \ldots, 12)$. The elements of each group are placed into the set of locations chosen from pos, which is driven by a provided secret bit-stream $M =' 01011010010110'$. Additional details are provided below.

The locations of the 4 elements in group G_1 should be chosen from the 12 locations in pos, so the amount of selections is $C(12, 4) = 495$. Since 495 options can be represented as $\lfloor \log_2 495 \rfloor = 8$ bits of secret information, we take the first 8 bits from the secret bit-stream, *i.e.*, $'01011010'$, as the decimal integer $m = 90$. Next, the $(m + 1) = 91 - th$ location, *i.e.* the locations "1,3,7,9", is chosen in dictionary sequence to arrange elements in G_1. Therefore, we determine the location of elements in first group to be $Ind[1]$, $Ind[3]$, $Ind[7]$, $Ind[9]$, *i.e.*, $Ind[1]$=$Ind[3]$=$Ind[7]$=$Ind[9]$=1. That implies the locations "1,3,7,9" will be employed to align the elements of G_1.

Likewise, arranging the $n = 4$ elements of G_2, 4 locations must be selected from the residual 8 locations in pos, so there are $C(8, 4) = 70$ to choose from. $\lfloor \log_2 70 \rfloor = 6$ bits of secret message can be concealed in 70 selections. Therefore, Then read the next 6 bits from the secret bit stream, *i.e.*, $'010110'$, as the decimal integer $m = 22$. Next, the $(m + 1) = 23 - th$ location choice *i.e.*, the locations "2,6,8,11", is chosen in dictionary sequence to order the four elements of G_2. Therefore, we determine the location of elements in G_2 *i.e.*, $Ind[2]$=$Ind[6]$=$Ind[8]$=$Ind[11]$=2. That implies the locations "2,6,8,11" will be employed to align the elements of G_2.

Driven on provided secret information M, the location Alignment array $Ind = (1, 2, 1, 3, 3, 2, 1, 2, 1, 3, 2, 3)$ can be obtained finally. Hence, the secret message M is coded to be an arrangement of the locations of these elements. The

below steps can be used to structure potential vectors, which is indicated by this array Ind.

Step (3): Latent vector construction. Arranging the arrayInd based on the locations it gets, the four elements in G_1 are placed in the "1-*th*, 3-*th*, 7-*th*, and 9-*th*" locations, the four elements in G_2 in the "2-*th*, 6-*th*, 8-*th*, and 11-*th*" locations, and the elements of groups G_3 are placed in the est of the locations in a random sequence. Get a N-dimensional vector by joining these aligned elements. As with a sum of N_T sampling elements, another $N_T - N$ elements are straight joined at the last of the vector to get the eventual N_T-dimensional latent vector z.

Step (4): picture generation. Randomly scramble elements of z using a private Key that is shared among the communicating participants, after constructing the N_T-dimensional latent vector z. Then the perturbation vector is projected onto the produced high-quality picture by the Stable NICE model. Eventually, the produced picture is employed as steganographic pictures for secret correspondence.

Under the recommended SM-IG programme, since the space of locational arrangements of N chosen elements is large enough for structuring high-dimensional potential vectors, the secret information in the structured vectors is efficiently encoded. In addition, the stable bijective-mapping between the structured vector and the produced picture can be achieved by the Stable NICE model. Therefore, the proposed SM-IG scheme has good efficiency and reversibility, which enables steganography with huge hiding volume and accurate secret message recovery.

Furthermore, by using the Stable NICE model, the constructed vector can be projected to a high-quality picture, and the message concealment can be achieved by producing fresh pictures instead of changing already existing ones. As a result, the proposed steganography method is ideally resistant to detection and unnoticeable, as demonstrated by the experiments in Sect. 5.

4.2 Reverse S2I Transformation for Information Extraction

Since message recovery is the reverse procedure of information concealment, the secret information can be obtained from the produced picture by implementing the reverse S2I transformation. The process of information extraction contains two major steps: latent vector extraction and secret message recovery.

Step (1): Latent vector recovery. At the receiver side, the Stable NICE model inversely maps the received picture into a latent vector. Then, the recovered latent vector $z^{'}$ can be obtained by the location sequence of the latent vector elements, which is recovered by the shared Key.

Step (2): Secret message extraction. Recover the potential vector with K and n known to both communicating parties and denote it as $z^{'}$, and regroup the first $N = Kn$ elements of $z^{'}$ using the method used in the information hiding phase. Get the location of elements of each group G_i, and according to get the location to determine its corresponding $No.$ in the location list, where the location list has $C(N-(i-1)n, n)$ possible options. Following step (2) of the information hiding phase, we can determine that the choice No. is $(m_i + 1)$ and the corresponding m_i is the decimal integer of relevant secret bit. Therefore, the corresponding selection number No. can be determined based on the location of n elements in each group G_i to get m_i, then m_i transformed to the relevant secret bit. Once all the bits are got, they are concatenated to extract the ultimate secret information $M^{'}$.

4.3 Hiding Capacity Analysis

The hiding capacity of SM-IG scheme is analyzed in this subsection. As mentioned in Sect. 4.1, for *i-th* group, $\lfloor \log_2 \omega_i \rfloor$-bit secret message can be embedded in the order of locations of its elements, where $\omega_i = C(N - (i-1)n, n)$. Therefore, the sum of bits that can be encoded as the location order for all groups is computed by

$$BN_{S2I} = \sum_{i=1}^{K-1} \lfloor \log_2 \omega_i \rfloor \tag{4}$$

It fulfils the formula.

$$\left(\sum_{i=1}^{K-1} \log_2 \omega_i \right) - K + 1 < \sum_{i=1}^{K-1} \lfloor \log_2 \omega_i \rfloor \leq \sum_{i=1}^{K-1} \log_2 \omega_i \tag{5}$$

where,

$$\sum_{i=1}^{K-1} \log_2 \omega_i = \log_2 \prod_{i=1}^{K-1} C(N - (i-1)n, n) \tag{6}$$

In accordance with the Stirling approximation, the below results can be obtained.

$$\begin{aligned} \sum_{i=1}^{K-1} \log_2 \omega_i &\approx \log_2 \left(\left(\frac{1}{\sqrt{2\pi}} \right)^{K-1} \frac{(Kn)^{Kn+\frac{1}{2}}}{n^{K(n+\frac{1}{2})}} \right) \\ &= -(K-1)\log_2 \sqrt{2\pi} + \left(Kn + \frac{1}{2} \right) \log_2 K - \frac{K-1}{2} \log_2 n \end{aligned} \tag{7}$$

Suppose that all the N_T sampled elements are selected and divided into N_T groups, *i.e.*, $N = N_T$ and $K = N_T$, with each group containing only $n = 1$ element for message hiding. The scale of the training pictures employed in the experiments are $28 \times 28 = 784$, and the latent vector N_T is constructed with the same dimension as the scale of these pictures, hence $N_T = 784$. Based on Eqs.

(5) and (7), the greatest BN_{S2I} is estimated to be approximately in the region of (5721,6504], and thus the largest number of bits hidden in per picture pixel (bpp) can theoretically be about $6504/784 \approx 8.3$, which is a very high level.

According to Eq. (6), larger K and n can result in better message hiding power but less accurate of message extraction. This will be analyzed in Sect. 5.2

5 Experiments

Within this section, firstly, we describe the experimental setup including training datasets, model training details, experimental environment, and assessment standards. Secondly, the influence of the parameters of the SM-IG scheme is analysed and discussed. That is, the effect of the number of element groups K and the number of elements n selected in each group. Thirdly, the validity of Stable NICE model is verified for the proposed method. Lastly, the performances of suggested SM-IG is evaluated and compared with the prior art in terms of resistance to detection and imperceptibility.

5.1 Experiment Settings

As the Stable NICE model consists of original NICE model and the rectification functions, we first train the original NICE model and then train the rectification functions.

Training Datasets: In [22], it is proven that the original NICE model trained on picture datasets can effectively generate high quality pictures. Therefore, in our experiments, we used two picture datasets, *i.e.*, MNIST dataset [23] and EMNIST dataset [24], to separately train the NICE model. The MNIST database is the dataset of 70K pictures, while the EMNIST database contains 145.6K pictures. The sizes of all the pictures in the two datasets are 28×28.

Model Training Details: By maximising the objective function defined in Eq. (6), the NICE model is trained. The learning rate of the training process is 10^{-3}. After 1500 training epochs on the two datasets, two trained NICE models are obtained, respectively. After obtaining the trained NICE models, we learn the parameters of the rectification functions defined in Eq. (7) and (8). To this end, we construct a set of 1000 latent vectors Z in the manner prescribed in Sect. 4.1 and then input the set into the NICE model trained on the MNIST database with or without the rectification function r_α to generate the two picture sets I and I_0, respectively. Then, the picture set I is reversely mapped by the trained NICE model to obtain the set of recovered latent vectors $Z^{'}$. Then, by minimizing the objective function defined in Eq. (9) with Genetic algorithm, we can determine the parameters, *i.e.*, $\alpha_1 = 3.0349$ and $\alpha_2 = -1.1655$.

Evaluation Criteria: In the experiments, we use the following evaluation criteria to assess the hiding power, extraction accuracy, detection resistance and imperceptibility of different steganography methods.

Hiding Capacity: The number of bits per pixel (*bpp*) is used to assess the information hiding ability of most picture steganography methods. It represents the number of secret bits hidden in per pixel. Also, we assess the hiding capacity IH_C by *bpp*.

Extraction Accuracy: The length of the initial secret bit stream may differ from the length of the recovered secret bit stream because of the hiding manner of suggested steganographic methods. Thus, we assess the precision of message extraction by calculating the Edit Distance (ED) between the original secret bit stream M and the extracted secret bit stream $M^{'}$. The accuracy rate is computed by

$$IE_A = 1 - \frac{ED(M, M^{'})}{\max[Len(M), Len(M')]} \tag{8}$$

where, $Len(M)$ and $Len(M^{'})$ are used to computing the lengths of bitstream M and $M^{'}$, respectively.

Anti-detectability: The following detection error rates were used to assess the performance of anti-detection against the steganalyser.

$$P_E = \min_{P_{FA}} \frac{1}{2}(P_{FA} + P_{MD}) \tag{9}$$

where P_{FA} is the false-alarm (FA) probability of steganalyzer and P_{MD} is the missed-detection (MD) probability of steganalyzer. A larger P_E implies that the steganography method has a higher resistance to detection against the steganography analyse [25].

5.2 Parameter Impacts

In the presented SM-IG scheme, there are two key parameters in order to construct the latent vectors: K and n, are the amount of groups of elements and the number of elements selected from each group, respectively. We evaluate the effect of the parameters on the performance of the suggested scheme in terms of hiding capacity and extraction accuracy. In this experiment, we adopt the trained Stable NICE model, which consists of the original NICE model trained on the MNIST dataset and the rectification functions trained on the set of 1000 constructed latent vector as described in previous subsection.

Figure 4 illustrates the effect of parameters on the proposed SM-IG. As shown in this diagram, the larger K and n are, obviously, the higher the hiding power. Since bigger K and n provide a bigger space for arranging the locations of the elements to structure the potential vectors, which allows to hide more secret bits in the produced pictures by the structured potential vector. Nevertheless, as K and n increase, the precision of message recovery decreases for the following

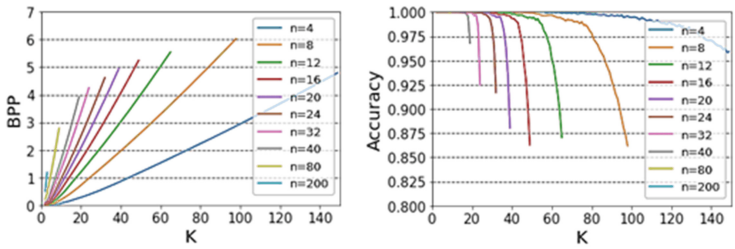

Fig. 4. The impacts of parameters K and n in the aspects of hiding capacity and extraction accuracy for SM-IG scheme.

reasons. Increases in K and n leads to more elements close to the group boundary, which makes the group $Nos.$ of these elements fragile. Thus, it is quite different for the group $Nos.$ of elements in the recovered latent vector and those in the raw latent vector, which significantly impacts the precise of message recovery.

According to Fig. 4, under the condition of perfect accuracy rate of information extraction, i.e., $IE_A = 1.0$, the hiding capacity of SM-IG can reach up to 4.3 *bpp*. As shown in Fig. 4, to reach the desired level of hiding capacity IH_C, there are several combinations of (K, n) to choose from. For instance, when SM-IG require the hiding capacity of $IH_C = 4$ *bpp*, the set of (K, n) are (23,32), (29,24), (33,20), (40,16), (50,12), (70,8) and (128,4). Moreover, with some hiding power, smaller K usually leads to higher extraction accuracy. Thus, the smallest K in the parameter combination, *i.e.*, (23,32), can be selected to obtain the desired hiding capacity $IH_C = 4$ *bpp*. In this way, it is possible to establish parameter combinations (K, n) for SM-IG that achieve a desired level of hiding power. These combinations can then be utilised in subsequent experiments.

5.3 Validity of Stable NICE Model

After setting the parameters in the above manner, we observe the validity of the designed Stable NICE model in this subsection. To this end, we test the accuracies of information extraction of SM-IG scheme with different levels of hiding payloads when using original NICE model or Stable NICE model. We denote the corresponding methods as NICE+SM-IG, S-NICE+SM-IG.

Figure 5 shows the accuracies of information extraction of the two methods with different levels of hiding payloads. From this figure, the accuracy of the method decreases as the hidden payload increases. It is clear that the accuracies of S-NICE+SM-IG is much higher than that of NICE +SM-IG. That indicates Stable NICE model can improve the information extraction accuracy significantly for SM-IG.

5.4 Performance Evaluation and Comparison

In this subsection, we evaluate and compare the performance of different steganographic methods in terms of accuracy, resistance to detection and impercepti-

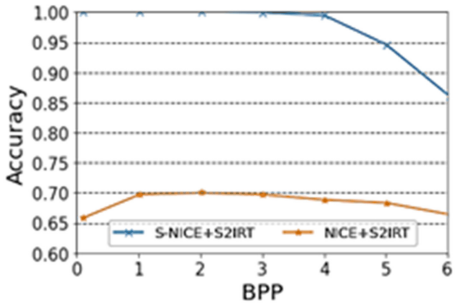

Fig. 5. Information extraction accuracy of SM-IG scheme when using original NICE model or Stable NICE model.

bility of information extraction under different hidden payloads. These methods include the following.

S-UNIWORD [26]: This well-known traditional steganography method is implemented based on a heuristically defined distortion function.

UT-6HPF-GAN [27]: It is a conventional steganographic approach and employs the distortion function learnt by the GAN.

SWE [16]]: The generative steganographic approach is named as steganography without embedding (SWE), where the secret information is encoded by fed into the DCGAN for stego-pictures generation.

S-NICE+SM-IG: In this approach, the proposed SM-IG scheme is implemented based on Stable NICE model for generative steganography.

In S-NICE+SM-IG, the stable NICE model trained on MNIST and that trained on EMNIST are employed to produce stego-pictures.

Information Extraction Accuracy: The message extraction precision (values of IE_A) of SWE, S-NICE+SM-IG with increasing hidden payload is shown in Table 1.

Table 1. The message extraction precision of stego-pictures of those methods with different hiding payloads.

Methods	Hiding payloads (*bpp*)				
	0.1	0.5	1.0	2.0	4.0
SWE	0.9983	0.8323	0.7096	0.6981	0.6844
S-NICE+SM-IG	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**0.9943**

Obviously, the proposed steganographic methods, *i.e.*, S-NICE+SM-IG, is much more accurate than other generative steganography methods*i.e.*, SWE, in terms of message extraction precision and under various hidden payloads. Also,

when the hiding payload were between 0.1 *bpp* and 4 *bpp*, the extraction accuracy rate remained at a very high-level ($IE_A \approx 1.0$). That is mainly because the approach has good efficiency and reversibility in secret-to-picture transformation process.

Therefore, the suggested generative steganography method can realize both hiding capacity (up to 4 *bpp*) and precise extraction of secret information (almost 100% accuracy rate).

Anti-detectability: In terms of estimating and comparing the anti-detection properties of different steganographic methods, the well-known steganalyzers, *i.e.*, SRM [15] and XuNet [3], used to check the existence of hidden messages in a steganographic picture. Among them, SRM is a steganalyser founded on a set of high-dimensional manual steganalysis features, and XuNet is a steganalyser founded on an improved CNN steganalysis structure. By using the two steganalyzers, the detection error rate P_E is calculated in the same way as described in Sect. 5.1 to assess the anti-detectability performance of various steganographic methods.

Table 2. The values of P_E of four steganographic methods with different hiding payloads.

	Methods	Hiding payloads(*bpp*)				
		0.1	0.5	1.0	2.0	4.0
SRM	S-UNIWORD	0.4229	0.1824	0.0493	-	-
	UT-6HPF-GAN	0.4414	0.2489	0.0615	-	-
	SWE	0.4987	-	-	-	-
	S-NICE+SM-IG	**0.5007**	**0.5011**	**0.4998**	**0.5002**	**0.5001**
XuNet	S-UNIWORD	0.4461	0.1925	0.0712	-	-
	UT-6HPF-GAN	0.4690	0.2971	0.0787	-	-
	SWE	0.4991	-	-	-	-
	S-NICE+SM-IG	**0.5001**	**0.4997**	**0.4986**	**0.5008**	**0.5002**

Table 2 shows the anti-detection behaviour of these methods for various hidden payloads (values of P_E) . From this table, we can draw a couple of observations.

For the low hiding payloads (0.1–0.5 *bpp*), we can observe that the generative steganographic approaches, *i.e.*, S-NICE+SM-IG and SWE, Outperforms traditional steganography methods by a wide margin, *i.e.*, S-UNIWORD and UT-6HPF-GAN. This is because these steganography generating methods do not need to modify the carrier picture and directly generate a new picture as a steganographic picture, and it is difficult for steganographers to discover the existence of hidden information.

For the high hiding payloads (larger than 0.5 *bpp*), it is obvious that the anti-detectability performances of S-NICE+SM-IG and SWE remain at high levels (P_E is about 0.5); On the contrary, the properties of the other methods inclusive of S-UNIWORD and UT-6HPF-GAN, is obviously degraded. In S-UNIWORD and UT-6HPF-GAN, the more secret information that is hidden in the cover picture, the more misrepresentation occurs, since secret information is hidden by altering the existing cover picture. As a result, steganographic analysers are more likely to easily detect these steganographic methods, especially at high hidden payloads. Note that for UT-6HPF-GAN and S-UNIWORD, the values of P_E will be null if the payload is greater than 1 *bpp*, as they are unable to get the hiding capacity greater than 1 *bpp*. Also, as shown in Table 1, when the hiding payload is larger than 0.1, the SWE is unable to extract the hidden information while the hidden load is greater than 0.1, and the corresponding P_E value of the SWE is null for following reasons. In SWE, the secret information is coded as a low-dimensional noise signal of DCGAN for generating a steganographic picture, and the information extractor is trained to extract the secret information from the steganographic picture. These make SWE difficult to achieve high hiding capacity with accurate information extraction. Thus, the high hiding power of SWE, S-UNIWORD and UT-6HPF-GAN comes at the cost of security performance or message extraction precision.

As a summary, the proposed steganographic approach, *i.e.*, S-NICE+SM-IG, can achieve desirable anti-detectability performances against the steganalyzers even at a very high hiding payload (4 *bpp*).

6 Conclusion

This work provides a new idea of generative steganography. A secret message-driven picture generation scheme (SM-IG) based on the designed Stable NICE model has been proposed for generative steganography. The proposed steganographic approach can obtain promising hiding capacity (over 4 *bpp*) while preserving the desired resistance to detection and unnoticeability, and dramatically outperforms state-of-the-art steganographic methods.

To conduct covert communication, many practical steganography tasks require hiding large amounts of message in an picture while preserving high extraction precision and desired resistance to detection and imperceptibility. These tasks can be well implemented by the proposed steganographic approach. Consequently, the proposed approach has important practical significance in the field of information hiding. As the NICE model is good at generating pictures, in the proposed steganographic approach, the secret messages are converted to the generated pictures. Going forward, we plan to extend the presented method by exploring additional flow models to generate other common types of pictures, for instance, face photos and landscape pictures for steganography.

Acknowledgements. This work is supported in part by the National Natural Science Foundation of China under Grant 62372125, Grant 61972205, Grant 62102462,

in part by the Guangdong Basic and Applied Basic Research Foundation under Grant no.2022A1515010108, in part by the Guangdong Natural Science Funds for Distinguished Young Scholar under Grant 2023B1515020041, and in part by the Collaborative Innovation Center of Atmospheric Environment and Equipment Technology (CICAEET) fund, China.

References

1. Filler, T., Judas, J., Fridrich, J.: Minimizing additive distortion in steganography using syndrome-trellis codes. IEEE Trans. Inf. Forensics Secur. **6**(3), 920–935 (2011)
2. Zhou, Z., Mu, Y., Wu, Q.: Coverless picture steganography using partial-duplicate picture retrieval. Soft. Comput. **23**(13), 4927–4938 (2019)
3. Wan, S., Gu, R., Umer, T., Salah, K., Xu, X.: Toward offloading internet of vehicles applications in 5G networks. IEEE Trans. Intell. Transp. Syst. **22**(7), 4151–4159 (2020)
4. Xu, J., et al.: Hidden message in a deformation-based texture. Vis. Comput. **31**(12), 1653–1669 (2015)
5. Saito, M., Matsui, Y.: Illustration2vec: a semantic vector representation of illustrations. In: SIGGRAPH Asia 2015 Technical Briefs, pp. 1–4 (2015)
6. Li, J., et al.: A generative steganography method based on WGAN-GP. In: Sun, X., Wang, J., Bertino, E. (eds.) ICAIS 2020. CCIS, vol. 1252, pp. 386–397. Springer, Singapore (2020). https://doi.org/10.1007/978-981-15-8083-3_34
7. Holub, V., Fridrich, J., Denemark, T.: Universal distortion function for steganography in an arbitrary domain. EURASIP J. Inf. Secur. **2014**(1), 1–13 (2014). https://doi.org/10.1186/1687-417X-2014-1
8. Wu, K.C., Wang, C.M.: Steganography using reversible texture synthesis. IEEE Trans. Picture Process. **24**(1), 130–139 (2014)
9. Li, S., Zhang, X.: Toward construction-based data hiding: from secrets to fingerprint pictures. IEEE Trans. Picture Process. **28**(3), 1482–1497 (2018)
10. Girshick, R.: Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1440–1448 (2015)
11. Cao, Y., Zhou, Z., Wu, Q.M.J., Yuan, C., Sun, X.: Coverless information hiding based on the generation of anime characters. EURASIP J. Image Video Process. **2020**(1), 1–15 (2020). https://doi.org/10.1186/s13640-020-00524-4
12. Jiang, W., Hu, D., Yu, C., Li, M., Zhao, Z.Q.: A new steganography without embedding based on adversarial training. In: Proceedings of the ACM Turing Celebration Conference-China, pp. 219–223 (2020)
13. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434 (2015)
14. Li, S., Zhang, X.: Toward construction-based data hiding: from secrets to fingerprint pictures. IEEE Transactions on picture Processing **28**(3), 1482–1497 (2018)
15. Fridrich, J., Kodovsky, J.: Rich models for steganalysis of digital pictures. IEEE Trans. Inf. Forensics Secur. **7**(3), 868–882 (2012)
16. Hu, D., Wang, L., Jiang, W., Zheng, S., Li, B.: A novel picture steganography method via deep convolutional generative adversarial networks. IEEE Access **6**, 38303–38314 (2018)

17. Peng, J., Sun, P., Zhang, L., Kuber, K., McLernon, D.: Timing synchronization for OFDMA femtocells in the presence of co-channel interference. In: 2012 8th International Wireless Communications and Mobile Computing Conference (IWCMC), pp. 1215–1220. IEEE (2012)
18. LeCun, Y.: The MNIST database of handwritten digits (1998). http://yann.lecun.com/exdb/mnist/
19. Xu, J., et al.: Hidden message in a deformation-based texture. Visual Comput. **31**(12), 1653–1669 (2015)
20. Arifianto, A., et al.: EDGAN: disguising text as picture using generative adversarial network. In: 2020 8th International Conference on Information and Communication Technology (ICoICT), pp. 1–6. IEEE (2020)
21. Jin, Y., Zhang, J., Li, M., Tian, Y., Zhu, H., Fang, Z.: Towards the automatic anime characters creation with generative adversarial networks. arXiv preprint arXiv:1708.05509 (2017)
22. Dinh, L., Krueger, D., Bengio, Y.: Nice: non-linear independent components estimation. arXiv preprint arXiv:1410.8516 (2014)
23. Le, Q., Mikolov, T.: Distributed representations of sentences and documents. In: International Conference on Machine Learning, pp. 1188–1196. PMLR (2014)
24. Cohen, G., Afshar, S., Tapson, J., Van Schaik, A.: Emnist: extending MNIST to handwritten letters. In: 2017 International Joint Conference on Neural Networks (IJCNN), pp. 2921–2926. IEEE (2017)
25. Luo, Y., Qin, J., Xiang, X., Tan, Y.: Coverless picture steganography based on multi-object recognition. IEEE Trans. Circuits Syst. Video Technol. **31**(7), 2779–2791 (2020)
26. Goodfellow, I., et al.: Generative adversarial nets. In: Advances in Neural Information Processing Systems, vol. 27 (2014)
27. Xu, G., Wu, H.Z., Shi, Y.Q.: Structural design of convolutional neural networks for steganalysis. IEEE Signal Process. Lett. **23**(5), 708–712 (2016)