Synchronized Detection and Recovery of Steganographic Messages with Adversarial Learning

Haichao Shi[1,2]*, Xiao-Yu Zhang[1]*, Shupeng Wang[1]**,
Ge Fu[3]**, and Jianqi Tang[3]

[1] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[3] National Computer Network Emergency Response Technical Team/Coordination Center of China, Beijing, China
{shihaichao,zhangxiaoyu,wangshupeng}@iie.ac.cn, fg@cert.org.cn,
tangjianqi1974@163.com

Abstract. In this work, we mainly study the mechanism of learning the steganographic algorithm as well as combining the learning process with adversarial learning to learn a good steganographic algorithm. To handle the problem of embedding secret messages into the specific medium, we design a novel adversarial module to learn the steganographic algorithm, and simultaneously train three modules called generator, discriminator and steganalyzer. Different from existing methods, the three modules are formalized as a game to communicate with each other. In the game, the generator and discriminator attempt to communicate with each other using secret messages hidden in an image. While the steganalyzer attempts to analyze whether there is a transmission of confidential information. We show that through unsupervised adversarial training, the adversarial model can produce robust steganographic solutions, which acts like an encryption. Furthermore, we propose to utilize supervised adversarial training method to train a robust steganalyzer, which is utilized to discriminate whether an image contains secret information. Extensive experiments demonstrate the effectiveness of the proposed method on publicly available datasets.

Keywords: Steganography · Steganalysis · Adversarial learning.

1 Introduction

Steganography aims to conceal a payload into a cover object without affecting the sharpness of the cover object. The image steganography is the art and science of concealing covert information within images, and is usually achieved by modifying image elements, such as pixels or DCT coefficients. Steganographic

---

* These authors contributed equally to this study and share the first authorship.
** Corresponding authors.

algorithms are designed to hide the secret information within a cover message such that the cover message appears unaltered to an external adversary. On the other side, steganalysis aims to reveal the presence of secret information by detecting the abnormal artefacts left by data embedding and recover the secret information of the carrier object. For a long period of time, many researchers have been involved in developing new steganographic systems. Meanwhile, the development of steganalytic tools are also gradually growing. Inspired by the ideas from adversarial learning, we model the relationship between these two aspects and optimize them simultaneously.

Learning methods have been widely studied in the computer vision community [19–29], among which adversarial learning is based on the game theory and is combined with unsupervised way to jointly train the model. Shi et al.[30] propose a novel strategy of secure steganography based on generative adversarial networks to generate suitable and secure covers for steganography, in which the decoding process was not considered. In this paper, we not only encode the secret messages into the images, but also decode them utilizing a discriminator network. Then, we utilize the steganalysis network to detect the presence of hidden messages. Through unsupervised training, the generator plays the role of a sender, which is utilized to generate steganographic images as real as possible. As the discriminator plays the role of a receiver, it not only differentiates between real images and generated images, but also extracts the secret messages. The steganalysis network, as a listener of the whole process, incorporates supervised learning with adversarial training to compete against state-of-the-art steganalysis methods.

In summary, this paper makes the following contributions:

- We incorporate the generative adversarial network with steganography, which proves to be a good way of encryption.
- We integrate unsupervised learning method and supervised learning method to train the generator and steganalyzer respectively, and receive robust steganographic techniques in an unsupervised manner.
- We also utilize the discriminative network to extract the secret information. Experiments are conducted on widely used datasets, to demonstrate the advantages of the proposed method.

The rest of the paper is structured as follows. In Section 2, we discuss the related work of steganography and adversarial networks. In Section 3, we elaborate the proposed method. Experiments are conducted in Section 4 to demonstrate the effectiveness of the proposed method. In Section 5, we draw conclusions.

2   Related work

2.1   Steganography

The image-based steganography algorithm can be split into two categories. The one is based on the spatial domain, the other is based on the DCT domain. In our

work, we mainly focus on the spatial domain steganography. The state-of-the-art steganographic schemes mainly concentrate on embedding secret information within a medium while minimizing the perturbations within that medium. On the contrary, steganalysis is to figure out whether there is secret information or not in the medium.

Least Significant Bit (LSB)[12] is one of the most popular embedding methods in spatial domain steganography. If LSB is adopted as the steganography method, the statistical features of the image are destroyed. And it is easy to detect by the steganalyzer. For convenience and simple implementation, the LSB algorithm hides the secret to the least significant bits in the given images channel of each pixel. Mostly, the modification of the LSB algorithm is called ±1-embedding. It randomly adds or subtracts 1 from the channel pixel, so the last bits would match the ones needed.

Besides the LSB algorithm, some sophisticated steganographic schemes choose to use a distortion function which is used for selecting the embedding localization of the image. This type of steganography is called the content-adaptive steganography. The minimization of the distortion function between the cover image $\mathcal{C}$ and the steganographic image $\mathcal{S}$ is usually required. These algorithms are the most popular and the most secure image steganography in spatial domain, such as HUGO (Highly Undetectable steGO)[4], WOW (Wavelet Obtained Weights)[2], S-UNIWARD (Spatial UNIversal WAvelet Relative Distortion)[3], etc.

$$d(\mathcal{C}, \mathcal{S}) = f(\mathcal{C}, \mathcal{S}) * |\mathcal{C} - \mathcal{S}| \tag{1}$$

where $f(\mathcal{C}, \mathcal{S})$ is the cost of modifying a pixel, which is variable in different steganographic algorithms.

HUGO is a steganographic scheme that defines a distortion function domain by assigning costs to pixels based on the effect of embedding some information within a pixel. It uses a weighted norm function to represent the feature space. WOW is another content-adaptive steganographic method that embeds information into a cover image according to textural complexity of regions. It is shown in WOW that the more complex the image region is, the more pixel values will be modified in this region. S-UNIWARD introduces a universal distortion function that is independent of the embedded domain. Despite the diverse implementation details, the ultimate goals are identical, i.e. they are all devoted to minimizing this distortion function to embed the information into the noise area or complex texture rather than the smooth image coverage area.

2.2   Adversarial Learning

In recent years, Generative Adversarial Networks (GANs) have been successfully applied to image generation tasks. The method that generative adversarial networks generate images can be classified into two categories in general. The first is mainly exploring image synthesis tasks in an unconditioned manner that generates synthetic images without any supervised learning schemes. Goodfellow

et al.[7] propose a theoretical framework of GANs and utilize GANs to generate images without any supervised information. However, the early GANs has somewhat noisy and blurry results and the gradient will be vanished when training the networks. Later, Radford et al.[15] propose a deep convolutional generative adversarial networks (DCGANs) for unsupervised representation. To solve the situation of gradient vanishing, WGAN[14] is proposed using the Wasserstein distance instead of the Jensen-Shannon divergence, to make the dataset distribution compared with the learning distribution from G.

Another direction of image synthesis with GANs is to synthesize images by conditioning on supervised information, such as text or class labels. The Conditional GAN[16] is one of the works that develop a conditional version of GANs by additionally feeding class labels into both generator and discriminator of GANs. Info-GAN[17] introduces a new concept, which divides the input noise z into two parts, one is the continuous noise signal that cannot be explained, and the other is called C. Where C represents a potential attribute that can be interpreted as a facial expression, such as the color of eyes, whether with glasses or not, etc. in the facial tasks. Recently, Reed et al.[18] utilize GANs for image synthesis using given text descriptions, which enables translation from character level to pixel level.

Adversarial learning has been applied to steganographic and cryptographic problems. In Abadis[1] work, they integrate two neural networks to an adversarial game to encrypt a secret message. In this paper, we are devoted to training a model that can learn a steganographic technique by itself making full use of the discriminator and recovering secret messages synchronously.

3   Adversarial Steganography

This section mainly discusses our proposed adversarial steganographic scheme. In section 3.1, we elaborate the network architecture of our model. Then, we formulate the problem of steganography and steganalysis to point out the foundation of the proposed steganographic scheme.

3.1   Network Architecture

Our model consists of three components, generator, discriminator and steganalyzer. The generator is used to generate steganographic images. Besides discriminating between real images and generated images, the discriminator is also utilized to extract the secret messages. And the steganalyzer is utilized to distinguish covers from steganographic images.

Generator. The generator receives the cover images and secret messages. Before learning the distribution of the images, the secret messages are first embedded into the cover images. The steganographic images embeded with secret messages are then put into the generator. We use a fully connected layer, which

can cover anywhere of the image not merely a fixed region. Then four fractionally-strided convolution layers, and finally a hyperbolic tangent function layer. The architecture is shown in Fig. 1.
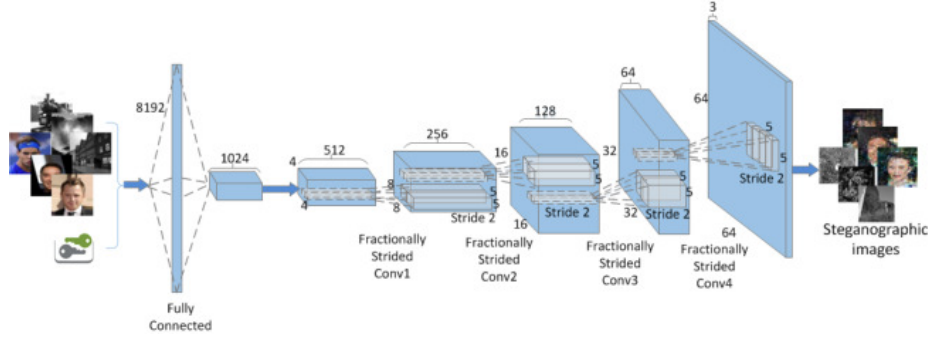

Fig. 1. The architecture of the generator.

Discriminator. The discriminator is mainly responsible for extracting the secret messages. Besides, it can also help to optimize the visual quality of the generated images. We use four convolutional layers and a fully connected layer. In detail, we additionally add a decoding function at the end of the network to extract the secret messages. The decoding function is acted as an interface between the steganographic images and the secret messages. It analyzes the modification of pixels in images, and recovers the secret messages. As shown in Fig. 2 is the architecture of discriminator.
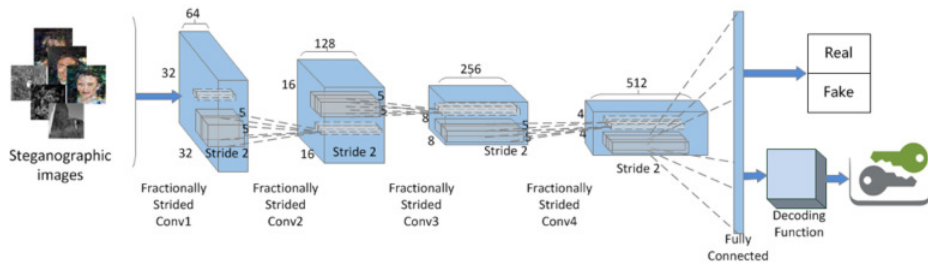

Fig. 2. The architecture of the discriminator.

Steganalyzer. The inputs of steganalyzer are both covers and steganographic images. Its architecture is similar to the discriminator, as is illustrated in Fig. 3.

With distinct, we first use a predefined high-pass filter to make a filtering operation, which is mainly for steganalysis. And then we use four convolutional layers. Finally, we use a classification layer, including a fully connected layer and a softmax classifier.
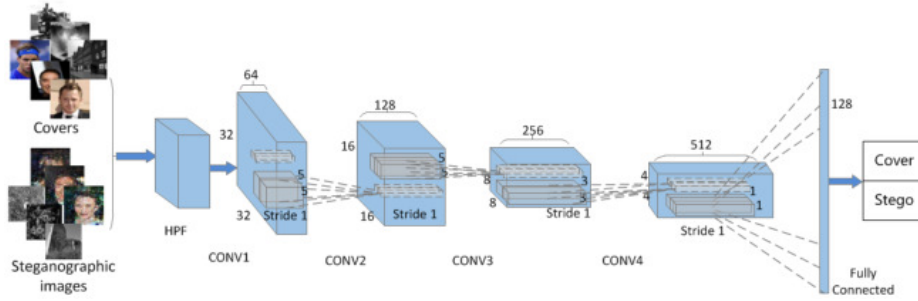

Fig. 3. The architecture of the steganalyzer.

3.2   Optimization Objective

Our training scheme includes three parties: generator, discriminator and steganalyzer. In this game, the generator conveys a secret medium to the discriminator, which the steganalyzer attempts to eavesdrop in the conveying process and find out whether there are some special information containing in the process. Generally speaking, the generator generates a steganographic image within the cover image embedded with a secret message. Then the generator passes it to the next network namely discriminator, concentrating on decoding and recovering the message. When an image is input to the steganalyzer, a confidence probability will be output by steganalyzer on how much likely the image is hidden with a secret message.

For the whole process of information conveying is transparent to the discriminator, so the decoding can be easily achieved. In the game of the three parties, the generator is trained to learn to produce a steganographic image such that the discriminator can decode and recover the secret message, and such that the steganalyzer can output a better probability whether it is a cover or a steganographic image.

The whole architecture can be depicted in Fig. 4, the generator receives a cover image, C, and a secret message, M. Combined with the above two inputs, it outputs a steganographic image, S_C, which is simultaneously given to the discriminator and steganalyzer. Once receiving a steganographic image, the discriminator decodes it and attempts to recover the message. For the game is under generative adversarial networks, the discriminator can also improve the
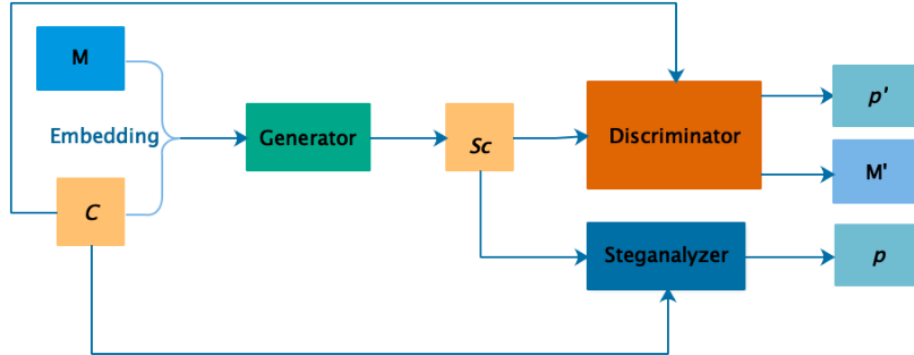
Fig. 4. The whole architecture of the training game. M and M' represents the secret message and extracted message from the steganographic image respectively, C represents the cover image, and S_C represents the steganographic image. p is the probability of classifying the covers and steganographic images. p' is the probability of classifying the real images and generated images.

visual quality of the generated images. In addition to the steganographic images, the cover images are also input into the steganalyzer, whose output is a probability of classifygin the steganographic images and covers.

Let $\omega_G, \omega_S, \omega_S$ denote the parameters of generator, discriminator and steganalyzer, respectively. We utilize $G(\omega_G, \mathrm{M}, C)$ represents generators output given an image C and secret message M, $D(\omega_D, S_C)$ represents discriminators output on steganographic images S_C, and $S(\omega_S, C, S_C)$ represents steganalyzers output on covers and steganographic images. Then we use L_G, L_D, L_S represent the loss of three networks respectively. The above formulations can be expressed as follows:

$$D(\omega_D, S_C) = D(\omega_D, G(\omega_G, \mathrm{M}, C)) \tag{2}$$

$$S(\omega_S, C, S_C) = S(\omega_S, C, G(\omega_G, \mathrm{M}, C)) \tag{3}$$

where $D(\omega_D, S_C)$ is output of the discriminator, $S_{(\omega_S, C, S_C)}$ is the output of the steganalyzer.

For the generator's loss, we correspond it with the other two networks. And we add constraints to normalize the loss.

$$L_G(\omega_G, \mathrm{M}, C) = \lambda_G \cdot d(C, S_C) + \lambda_D \cdot L_D(\omega_D, S_C) + \lambda_S \cdot L_S(\omega_S, C, S_C) \tag{4}$$

where $d(C, S_C)$ is the Euclidean distance between the covers and the steganographic images. And λ_G, λ_D, λ_S are the calculation loss weights of the generator, discriminator and steganalyzer respectively. These variables are varying from $[0, 1]$.

Also, we set discriminator's loss to be the norm form. Here, we use the ℓ_2 norm.

$$L_D(\omega_G, \omega_D, M, C) = ||M, D(\omega_D, S_C)||_2 = ||M, D(\omega_D, G(\omega_G, M, C))||_2 = ||M, M'||_2 \tag{5}$$

where the form $||M, M'||_2$ is the distance between M and M'.

And we set steganalyzer's loss to be binary cross-entropy of logistic regression.

$$L_S(\omega_G, \omega_S, C, S_C) = -\frac{1}{n}\sum_x \ln S(\omega_S, C, S_C) + (1-y)\ln(1 - S(\omega_S), C, S_C) \tag{6}$$

where $y = 1$ if $x = C$, which means that the images are classified as covers, $y = 0$ if $x = S_C$, which means the steganographic images are identified. We apply adversarial learning techniques to discriminative tasks to learn the steganographic algorithm. Under unsupervised adversarial training, the model can produce steganographic techniques, which acts like an encryption.
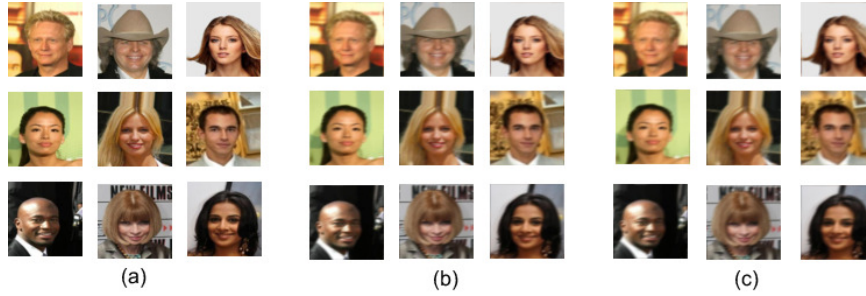

Fig. 5. The covers and steganographic images of CelebA dataset. (a) represents the covers, (b) represents the steganographic images with the embedding rates of 0.1bpp, (c) represents the steganographic images with the embedding rates of 0.4bpp.

4   Experiments

Experiments are conducted on two image datasets: celebrity faces in the wild (CelebA)[8] and BossBase dataset. We perform the center crop of size 64×64 for both two datasets. For the choice of M, we concatenate a random paragraph message to generalize our model to random message, with each sample of each dataset. And we consider different kinds of embedding rates, which vary from 0.1bpp (bits per pixel) to 0.4bpp. Generally, most steganographic algorithms can successfully hide secret messages approximately at 0.4bpp. All experiments are performed in TensorFlow[9], on a workstation with a Titan X GPU.
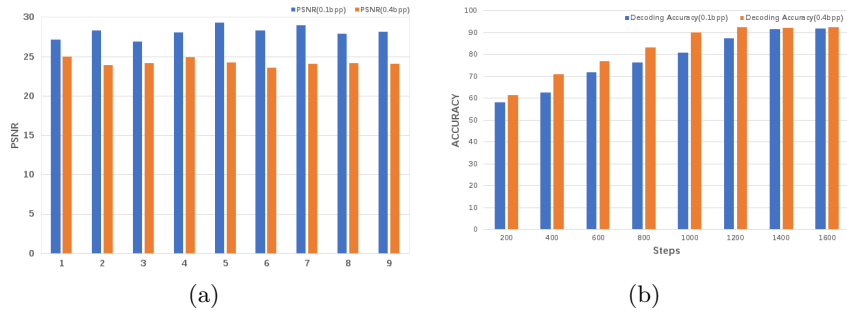
Fig. 6. (a) represents the contrast of PSNR of embedding rates at 0.1bpp and 0.4bpp, where the number in horizontal axis represents the numerical order of images in Fig. 5 of each part from left to right, up to bottom. (b) represents the contrast of decoding accuracy between 0.1bpp and 0.4bpp.

4.1   CelebA dataset

CelebA dataset contains 202,599 images of celebrity faces from 10,177 unique identities. We train the model using RMSProp optimization algorithm with the learning rate of 2×10^{-4}. At each batch we jointly train the generator, discriminator and steganalyzer.

From the results, we can see that there is no noticeable image quality decrease between 0.1bpp and 0.4bpp. As is shown in Fig. 5. And we also calculate the PSNR (Peak Signal to Noise Ratio) to evaluate the steganographic effect. As shown in Fig. 6.

From the charts, we can see that the discriminator can hardly decode the secret messages in the first few rounds of training. For the visual quality of the generated images is low and the discriminator is randomly guessing the messages essentially. After 800 training steps, the discriminator can decode the messages correctly with an average success of 90% at 0.4bpp. After 1,000 training steps, the discriminator approaches to convergence gradually and decodes the messages with success rate above 95%. We can also conclude that the discriminator performs better under embedding rate at 0.4bpp than 0.1bpp when decoding the messages.

4.2   BossBase dataset

In addition to the experiments on the CelebA dataset, we also train our model on the BossBase dataset, which is a standard steganography dataset consisting of 10,000 grayscale images of various scenes. For the images from this dataset do not come from a single distribution, so the dataset can perform worse than the experiments on the CelebA dataset.

Qualitative results on BossBase are shown in Fig. 7. Contrast of PSNR and decoding accuracy at 0.1bpp and 0.4bpp is shown in Fig. 8. While the decoding rates suffers slightly from the image quality, the discriminator is still able to

learn to decode the messages successfully. This also implies the effectiveness of the generator, which is able to learn a good steganographic algorithm.

We can conclude from the two groups of experiments that for the datasets with distinct distribution, different embedding rates can influence the similarity between the original images and the embedded images.

(a)   (b)   (c)

Fig. 7. The covers and steganographic images of BossBase dataset. (a) represents the covers, (b) represents the steganographic images with the embedding rates of 0.1bpp, (c) represents the steganographic images with the embedding rates of 0.4bpp.
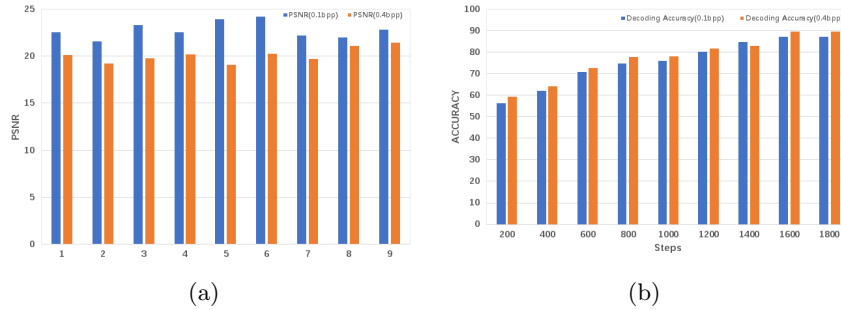

(a)   (b)

Fig. 8. (a) represents the contrast of PSNR of embedding rates at 0.1bpp and 0.4bpp, where the number in horizontal axis represents the numerical order of images in Fig. 7 of each part from left to right, up to bottom. (b) represents the contrast of decoding accuracy between 0.1bpp and 0.4bpp.

4.3   Comparison with the state-of-the-arts

We train in such a way that the steganalyzer can detect the covers and steganographic images with low rates. Thus, on the basis of the experiment, we addition-

ally conduct another experiments compared with the state-of-the-art steganographic algorithms, which shows competitive results. Note that Steganalyzer* denotes the proposed steganalysis method, and Generator* denotes the proposed steganography method.

Table 1. Accuracy of distinguishing between cover and steganographic images for the GNCNN and Steganalyzer* at an embedding rate of 0.4bpp.

Steganographic Algorithm	CelebA		BossBase	
	Steganalyzer		Steganalyzer	
	GNCNN	Steganalyzer*	GNCNN	Steganalyzer*
HUGO	0.93	0.91	0.72	0.68
S-UNIWARD	0.91	0.89	0.75	0.71
Generator*	0.94	0.92	0.83	0.82

For both datasets, we compare our scheme against steganographic algorithms including HUGO[4] and S-UNIWARD[3]. As shown in Table 1, Steganalyzer* performs competitively with other methods. By training Steganalyzer* in a supervised fashion, the experiments show that it has the capacity to become a strong steganalyzer, competing against established techniques like GNCNN[10], and so is a good choice for the steganalyzer. Furthermore, for both the CelebA and BossBase datasets, we use 10,000 samples and split them in half, to create a training set and a test set. We then train the Generator* on the training set and test the model on the testing set. Following this, additional 5,000 steganographic images are created utilizing each steganography algorithm. Thus, the cover set and steganographic images set are all consisting of 5,000 images respectively.

As demonstrated in the experiments, Steganalyzer* performs competitively against the GNCNN, and the Generator* also performs well against other steganographic techniques. The experimental results show that on the one hand, the generated images are good at visual quality. On the other hand, the generated images are harder to detect, which shows the security of the generated images as covers.

5   Conclusion

In this paper, an adversarial steganography architecture with generative adversarial networks is proposed. On the basis of generative adversarial networks, we leverage the adversarial structure to form an effective steganographic method. Encouraging results are received from experiments conducted on the widely used datasets in comparison with several state-of-the-art methods. It is also worth studying the adaption of the adversarial steganography architecture in adaptive steganographic algorithm. Furthermore, it is interesting to resort the networks to a game-theoretic formulation when we characterize the interplay between the steganographer and the steganalyst.

6   Acknowledgements

References

1. M. Abadi and D. G. Andersen. Learning to protect communications with adversarial neural cryptography. CoRR, abs/1610.06918, 2016.
2. V. Holub and J. J. Fridrich. Designing steganographic distortion using directional filters. In 2012 IEEE International Workshop on Information Forensics and Security, WIFS 2012, Costa Adeje, Tenerife, Spain, pages 234-239, 2012.
3. V. Holub, J. J. Fridrich, and T. Denemark. Universal distortion function for steganography in an arbitrary domain. EURASIP J. Information Security, 2014.
4. T. Pevn, T. Filler, and P. Bas. Using high-dimensional image models to perform highly undetectable steganography. In Information Hiding-12th International Conference, IH 2010, Calgary, AB, Canada, Revised Selected Papers, pages 161-177, 2010.
5. G. Xu, H.Wu, and Y. Shi. Structural design of convolutional neural networks for steganalysis. IEEE Signal Process. Lett., 23(5):708-712, 2016.
6. G. Xu, H.Wu, and Y. Q. Shi. Ensemble of cnns for steganalysis: An empirical study. In Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security, IH&MMSec 2016, Vigo, Galicia, Spain, pages 103-107, 2016.
7. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D.Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, Advances in Neural Information Processing Systems 27, pages 2672-2680. Curran Associates, Inc., 2014.
8. Z. Liu, P. Luo, X.Wang, and X. Tang. Deep learning face attributes in the wild. In 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, pages 3730-3738, 2015.
9. M. Abadi, A. Agarwal, P. Barham, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. CoRR, abs/1603.04467, 2016.
10. Y. Qian, J. Dong, W. Wang, and T. Tan. Deep learning for steganalysis via convolutional neural networks. In Media Watermarking, Security, and Forensics 2015, San Francisco, CA, USA, Proceedings, page 94090J, 2015.
11. M. Browne, S. S. Ghidary, and N. M. Mayer. Convolutional neural networks for image processing with applications in mobile robotics. In Speech, Audio, Image and Biomedical Signal Processing using Neural Networks, pages 327-349. 2008.
12. J. Mielikainen. LSB matching revisited. IEEE Signal Process. Lett., 13(5):285-287, 2006.
13. A. D. Ker. Resampling and the detection of LSB matching in color bitmaps. In Security, Steganography, and Watermarking of Multimedia Contents VII, San Jose, California, USA, Proceedings, pages 1-15, 2005.
14. M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In D. Precup and Y. W. Teh, editors, Proceedings of the 34th International Conference on Machine Learning, volume 70 of Proceedings of Machine Learning Research, pages 214-223, International Convention Centre, Sydney, Australia, 2017. PMLR.

15. A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. CoRR, abs/1511.06434, 2015.

16. A. Odena, C. Olah, and J. Shlens. Conditional image synthesis with auxiliary classifier gans. In Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, pages 2642-2651, 2017.

17. X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, Barcelona, Spain, pages 2172-2180, 2016.

18. S. E. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative adversarial text to image synthesis. In Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, pages 1060-1069, 2016.

19. X. Zhang, H. Shi, X. Zhu, P. Li. Active semi-supervised learning based on self-expressive correlation with generative adversarial networks. Neurocomputing, 2019. doi:10.1016/j.neucom. 2019.01.083.

20. X. Zhang. Simultaneous optimization for robust correlation estimation in partially observed social network. Neurocomputing, 205:455-462, 2016.

21. X. Zhang, S. Wang, X. Yun. Bidirectional active learning: a two-way exploration into unlabeled and labeled dataset. IEEE Transactions on Neural Networks and Learning Systems, 26(12), 3034-3044, 2015.

22. X. Zhang, S. Wang, X. Zhu, X. Yun, G. Wu and Y. Wang. Update vs. upgrade: modeling with indeterminate multi-class active learning. Neurocomputing, 162, 163-170, 2015.

23. X. Zhang. Interactive Patent classification based on multi-classifier fusion and active learning. Neurocomputing, 127, 200-205, 2014.

24. X. Zhang, H. Shi, C. Li, K. Zheng, X. Zhu and L. Duan. Learning transferable self-attentive representations for action recognition in untrimmed videos with weak supervision. In Proceedings of the 33nd AAAI Conference on Artificial Intelligence, 1-8, 2019.

25. X. Zhang, C. Xu, J. Cheng, H. Lu and S. Ma. Effective annotation and search for video blogs with integration of context and content analysis. IEEE Transactions on Multimedia, 11(2), 272-285, 2009.

26. Y. Liu, X. Zhang, X. Zhu, Q. Guan and X. Zhao. ListNet-based object proposals ranking. Neurocomputing, 267, 182-194, 2017.

27. C. Li, Q. Liu, J. Liu, and H. Lu. Ordinal distance metric learning for image ranking. IEEE Transactions on Neural Networks, 26(7):1551-1559, 2015.

28. C. Li, X. Wang, W. Dong, J. Yan, Q. Liu, and H. Zha. Joint active learning with feature selection via cur matrix decomposition. IEEE Transactions on Pattern Analysis and Machine Intelligence, pages 1-1, 2018.

29. C. Li, F. Wei, W. Dong, X. Wang, Q. Liu, and X. Zhang. Dynamic structure embedded online multiple-output regression for streaming data. IEEE Transactions on Pattern Analysis and Machine Intelligence, 41(2):323-336, 2019.

30. H. Shi, J. Dong, W. Wang, Y. Qian, and X. Zhang. SSGAN: secure steganography based on generative adversarial networks. In Advances in Multimedia Information Processing - PCM 2017 - 18th Pacific-Rim Conference on Multimedia, Harbin, China, Revised Selected Papers, Part I, pages 534-544, 2017.