PATCH STEGANALYSIS: A SAMPLING BASED DEFENSE AGAINST ADVERSARIAL STEGANOGRAPHY

Chuan Qin, Na Zhao, Weiming Zhang, Nenghai Yu*

School of Cyber Science and Technology, University of Science and Technology of China;
Key Laboratory of Electromagnetic Space Information, Chinese Academy of Sciences.

ABSTRACT

In recent years, the classification accuracy of CNN (convolutional neural network) steganalyzers has rapidly improved. However, as general CNN classifiers will misclassify adversarial samples, CNN steganalyzers can hardly detect adversarial steganography, which combines adversarial samples and steganography. Adversarial training and preprocessing are two effective methods to defend against adversarial samples. But literature shows adversarial training is ineffective for adversarial steganography. Steganographic modifications will also be destroyed by preprocessing, which aims to wipe out adversarial perturbations. In this paper, we propose a novel sampling based defense method for steganalysis. Specifically, by sampling image patches, CNN steganalyzers can bypass the sparse adversarial perturbations and extract effective features. Additionally, by calculating statistical vectors and regrouping deep features, the impact on the classification accuracy of common samples is effectively compressed. The experiments show that the proposed method can significantly improve the robustness against adversarial steganography without adversarial training.

Index Terms— Steganalysis, adversarial steganography, sampling, image patch.

1. INTRODUCTION

Steganography is a covert communication technique. Modern steganography is based on the minimal cost model [1]. It formulates the steganography problem as source coding with fidelity constraint. As STC [2] and SPC [3] have realized coding performance that approximates the rate cost bound, steganographic research has focused on designing cost functions [4, 5, 6].

Steganalysis aims to detect whether an image is embedded with secret messages. Early steganalysis relied on complex and high-dimensional handcrafted features [7, 8]. It mainly takes two steps to extract handcrafted features: calculating residual maps and extract statistical features. To highlight subtle steganographic signals, it utilizes high-pass filter banks to calculate residuals. Co-occurrence matrices or histograms are then taken to generate high-dimensional features. Combined with traditional machine learning tools [9, 10], handcrafted feature steganalyzers perform well in detecting stego images. Recently, CNN (convolutional neuron network) significantly improves the classification accuracy of steganalysis. Since YeNet [11], the performance advantage of CNN steganalyzers [12, 13] over handcrafted feature steganalyzers has grown consistently.

CNN steganalyzers, like other CNN models for image classification tasks, are challenged by adversarial samples [14, 15, 16]. Adversarial samples are crafted by adding subtle and imperceptible noise to the natural images. They cause target CNN models to output incorrect results. Adversarial steganography [17, 18] can accomplish both the deception of target CNN steganalyzers and the delivery of secret messages. Currently, there mainly two approaches, cover enhancement and cost adjustment. Zhang et al. [17] proposed to iteratively enhance cover images until they are still classified as cover after being embedded with secret messages. Tang et al. [18] proposed to adjust the costs of part image elements, which forces the directions of steganographic modifications the same as the gradients towards cover class. The steganographic modifications on the elements of adjusted costs are encoded with part of secret messages and function as adversarial perturbations at the same time.

In general image classification tasks, preprocessing [19, 20] and adversarial retraining [15, 21] are considered to be the most effective methods for defending against adversarial samples. However, Bernard et al. [22] and Tang et al. [18] found that adversarial retraining was not effective in defending against adversarial steganography. Preprocessing aims to wipe out adversarial perturbations via image transformations [19] or denoising [20]. Unlike general classification tasks that focus on semantic information of images, steganalysis aims to detect steganographic modifications that are as subtle as adversarial perturbations. They will also be wiped out during the preprocessing. So preprocessing is generally considered unsuitable for the steganalysis.

In this paper, we notice that adversarial perturbations are sparse while steganographic modifications are scattered over the entire image. Therefore, we propose to sample image patches uniformly from regions of varying modification probabilities. It can bypass sparse adversarial perturbations and forces CNN steganalyzers to scatter attentions over varying regions from the whole image. In addition, the inter-patch correlations are considered to improve the detection ability. Specifically, we calculate dimension-wise statistical vectors and regroup the deep features extracted from sampled patches and assign them to different base learners. The experiment shows that the proposed method effectively improve the robustness of CNN steganalyzers against adversarial steganography without adversarial training. At the same time, the drop of the detection accuracy on common samples (cover and conventional stego images) is marginal.

This work was supported in part by the Natural Science Foundation of China under Grant 62002334, 62072421 and 62121002, and by Anhui Science Foundation of China under Grant 2008085QF296, and by Anhui Initiative in Quantum Information Technologies under Grant AHY150400.

E-mail:{qc94,znzhaona}@mail.ustc.edu.cn,
{zhangwm,ynh}@ustc.edu.cn.

* Corresponding author.

2. RELATED WORK

ADV-EMB [18] generates adversarial stego images by forcing the embedding cost fit the gradient sign. It divides the elements of the cover image into two disjoint groups, common group and adjustable group. The embedding costs in common group are defined by the base cost function, such as UNIWARD [4], HILL [5], UERD [6], and etc. The embedding costs in adjustable group are adjusted based on the gradient map of the stego image generated in the last iteration:

$$q_{i,j}^{+} = \begin{cases} \rho_{i,j}^{+}/\alpha, & \text{if } \eta_{i,j} < 0, \\ \rho_{i,j}^{+}, & \text{if } \eta_{i,j} = 0, \\ \rho_{i,j}^{+} \cdot \alpha, & \text{if } \eta_{i,j} > 0, \end{cases} \tag{1}$$

$$q_{i,j}^{-} = \begin{cases} \rho_{i,j}^{-}/\alpha, & \text{if } \eta_{i,j} > 0, \\ \rho_{i,j}^{-}, & \text{if } \eta_{i,j} = 0, \\ \rho_{i,j}^{-} \cdot \alpha, & \text{if } \eta_{i,j} < 0, \end{cases} \tag{2}$$

where the gradient value, base cost value and adjusted cost value at the element with position index i, j are denoted as $\eta_{i,j}$, $\rho_{i,j}$ and $q_{i,j}$.

3. METHOD

3.1. The Sparsity of Adversarial Perturbations

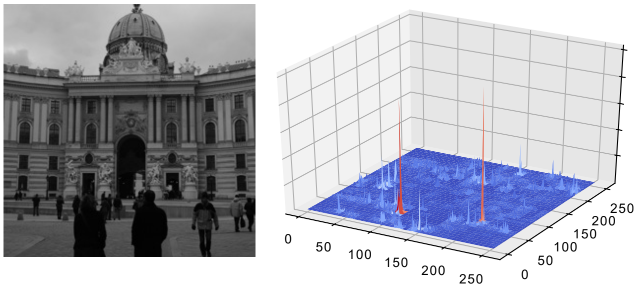

Fig. 1: The S-UNIWARD [4] stego image and its corresponding gradient value map with SRNet [12] being the target.

Under the condition of deceiving the target CNN steganalyzer, adversarial steganography tries to minimize introduced adversarial perturbations. ADV-EMB [18] can deceive XuNet [23] under payload 0.4 bpnzAC (bit per non-zero AC) by only adjusting modification costs of averaging about 16% image elements. It is clear that the adversarial perturbations of ADV-EMB is sparse.

In addition, in a stego image, the gradient values of the vast majority of pixels are low, whereas a small number of pixels have exceptionally high gradient values. The 3D graph of the gradient map of a S-UNIWARD [4] stego image are shown in Fig. 1. It indicates that the perturbations on such a small group of pixels can effectively alter the predictions, while perturbing other pixels could hardly deceive CNN steganalyzers.

Hence, one can conclude that the adversarial perturbations in steganalysis are sparse.

3.2. Patch Steganalysis

Since the adversarial perturbations are sparse, sampling a group of image patches can bypass most of them. Even some are sampled, the quantity of adversarial perturbations can hardly be sufficient to deceive CNN steganalyzers. Meanwhile, a defense method against adversarial steganography should reduce the defects on classifying common samples. So, including the deep features extracted from the sampled patches, a group of statistical vectors are calculated. Then all the features are regrouped and assigned to several base learners. The final predictions on input images are based on the majority voting of these base learners. The complete process of patch steganalysis is shown in Fig. 2.

3.2.1. Sampling patches

Through defining modification costs [4, 5, 6], pixels in textured regions are more likely to be modified than those in smooth regions. But the dense modifications in textured regions and sparse modification in smooth regions are equally important for steganalysis. Hence, we propose to sample image patches from regions with varying modification probabilities.

Specifically, the input image is first segmented with overlap to produce a number of candidate patches of the same size. As shown in the left side of Fig. 2, the candidate patches are of size $b \times b$, and the sampling stride is s.

By defining texture complexity, we can predict the modification probabilities of pixels. Thus, each image patch $\boldsymbol{X}_k$ obtains a predicted modification probability matrix $\boldsymbol{P}_k$. All the candidate patches are sorted according to the sum of elements in the modification probability matrix:

$$\begin{aligned} &[\boldsymbol{X}_{r_1},\ \boldsymbol{X}_{r_2},\ \ldots,\ \boldsymbol{X}_{r_n}],\ [r_1,\ r_2,\ \cdots,\ r_n] \\ &= \text{sort}([\boldsymbol{X}_1,\ \boldsymbol{X}_2,\ \ldots,\ \boldsymbol{X}_n],\ [\boldsymbol{P}_1,\ \boldsymbol{P}_2,\ \ldots,\ \boldsymbol{P}_n]), \end{aligned} \tag{3}$$

where X_{r_k} represents the image patch with the r_k-th high modification probability sum. Then the image patches are evenly divided into g groups:

$$\boldsymbol{G}_i = [\boldsymbol{X}_{r_{\text{seq}(i)+1}},\ \boldsymbol{X}_{r_{\text{seq}(i)+2}}, \ldots, \boldsymbol{X}_{r_{\text{seq}(i+1)}}], \tag{4}$$

$$\text{seq}(i) = (i-1) \cdot \lceil \frac{n}{g} \rceil, \tag{5}$$

Lastly, a representative patch is chosen randomly from each group. Accordingly, the image patches of either high or low predicted modification probabilities are sampled.

3.2.2. Feature fusion

The penultimate layer outputs of the target steganalyzer is the deep features in this paper. The deep features are extracted from the representative patches first.

Previous steganalytic methods utilized the whole image to extract global features. It constructs direct or indirect correlations among all regions. But, such correlations among representative patches are lost in the sampling process. Hence, we calculate the statistical vectors of all deep features:

$$\begin{aligned} \boldsymbol{f}_{\text{min}} &= [\min(\boldsymbol{f}_1), \min(\boldsymbol{f}_2), ..., \min(\boldsymbol{f}_d)], \\ \boldsymbol{f}_{\text{max}} &= [\max(\boldsymbol{f}_1), \max(\boldsymbol{f}_2), ..., \max(\boldsymbol{f}_d)], \\ \boldsymbol{f}_{\text{mean}} &= [\text{mean}(\boldsymbol{f}_1), \text{mean}(\boldsymbol{f}_2), ..., \text{mean}(\boldsymbol{f}_d)], \\ \boldsymbol{f}_{\text{std}} &= [\text{std}(\boldsymbol{f}_1), \text{std}(\boldsymbol{f}_2), ..., \text{std}(\boldsymbol{f}_d)], \end{aligned} \tag{6}$$

where $\boldsymbol{f}_i$ represents i-th dimension of deep features of all image patches, and $\min(\cdot)$, $\max(\cdot)$, $\text{mean}(\cdot)$ and $\text{std}(\cdot)$ mean calculating the minimum, maximum, average and standard deviation respectively, and $\boldsymbol{f}_{min}$ represents the minimal value vector, and vise versa.

Then, including the statistical vectors, all features are regrouped and assigned to several base learners. This process is implemented by ensemble classifier [9].

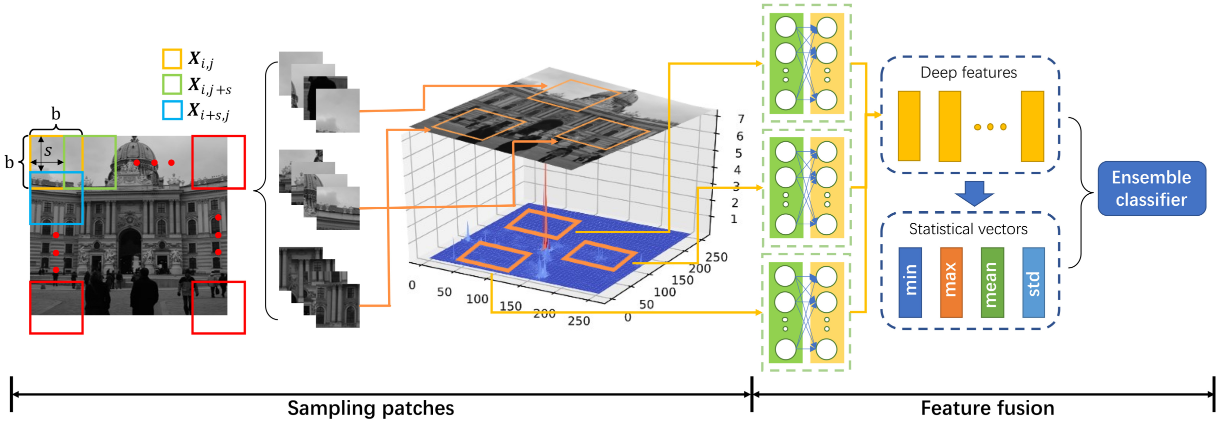

Fig. 2: Patch steganalysis. The image patches are sampled randomly based on estimated modification probabilities. The image patch with $x_{i,j}$ as the left-up starting point is denoted as $\boldsymbol{X}_{i,j}$. Its adjacent patches in the same row and column are $\boldsymbol{X}_{i,j+s}$ and $\boldsymbol{X}_{i+s,j}$. The extracted deep features and the statistical vectors are sent to the ensemble classifier, through which the model outputs its predictions.

4. EXPERIMENTS

4.1. Setups

4.1.1. Datasets

The experiments in this paper are conducted on two widely studied datasets, BOSSBase 1.01 [24] and BOWS2 [25]. Each contains $10,000$ grayscale images of 512×512. To match the settings of previous works, the original images are resized to 256×256 by **imresize()** of MatLab.

4.1.2. Hyperparameters

The value of stride s, patch size b and patch number p are set 48, 80 and 15 respectively. To improve the representative ability of deep features extracted from the patches, the target CNN steganalyzer is trained on cropped cover and conventional stego images of $b \times b$. The optimization process of these parameters is detailed in Section 4.4.3.

4.1.3. Target models and attack methods

SRNet [12] as one of the state-of-the-art CNN steganalyzers is adopted as the target model in this paper. The classic adversarial steganographic method ADV-EMB [18] is selected to evaluate robustness of steganalyzers.

In this paper, we assume that the steganographer has access to the defense approach and conduct adaptive attacks. Specifically, the steganographer calculates all the element gradients with reference to the model trained on cropped images of $b \times b$ and generate adaptive adversarial stego images targeting the patch steganalysis. Adversarial training augments training set with adversarial stego images and update the model weights. The adaptive attack targeting it crafts adversarial stego images based on the updated model weights.

4.2. Robustness Improvements

The proposed method aims to improve the robustness of CNN steganalyzers against adversarial steganography. Even without adversarial training, as shown in Table 1, patch defense can effectively improve the detection accuracy of the CNN steganalyzer against ADV-EMB under all tested payloads. The most significant improvement is 66.22% under payload 0.4 bpp (bit per pixel). The average improvement across all the payloads is 44.47%. Furthermore, compared with adversarial trained SRNet, the advantages of the proposed method is significant. First, as shown in Table 1, under adaptive attacks, patch steganalysis is notably more robust than adversarial trained SRNet. Second, adversarial training is quite time-consuming. The steganalyst must craft adversarial stego images and retrain models. While patch steganalysis is free from such process.

Table 1: The detection accuracy ($\%$) on adversarial steganography. Adversarial trained SRNet is denoted as “SRNet-adv” and patch steganalysis is denoted as “patch”.

Payload	0.1	0.2	0.3	0.4	0.5
SRNet	11.88	11.22	10.08	5.00	5.22
SRNet-adv	26.18	23.66	29.10	29.08	27.50
Patch	36.72	43.44	48.88	71.22	65.48

4.3. Practical detection performances

Table 2: Practical classification performances (detection accuracy with 5% false alarm)

Payload	0.1	0.2	0.3	0.4	0.5
SRM+EC	15.97	21.64	25.86	32.39	40.86
SRNet	17.16	31.52	39.42	44.16	47.45
Patch	**18.50**	**34.76**	**55.02**	**61.38**	**67.40**

In reality, stego images are the mixture of the conventional and the adversarial. Moreover, steganalysis prioritizes compressing false alarms over missed detection. Hence, in this section, we compare the detection accuracies of steganalyzers on the mixture of $1 : 1$ conventional and adversarial stego images at a fixed false alarm rate of 5%.

Due to missed detection of adversarial stego images, as shown in Table 2, the performance gap between SRNet and SRM+EC [7, 9] is narrow. The improvement brought by the patch defense on the detec-

Table 3: The performance comparison between the patch defense with and without statistical vectors.

Acc on		0.1	0.2	0.3	0.4	0.5
Mixture at $P_{FA} = 5\%$	Without stat vec	12.26	**38.47**	43.32	53.20	60.72
	With stat vec	**18.50**	34.76	**55.02**	**61.38**	**67.40**

Table 4: The performance comparison between the patch defense of training image size 256 × 256 and 80 × 80.

Acc on	Size	0.1	0.2	0.3	0.4	0.5
ADV-EMB	256	**53.14**	**44.68**	43.90	67.36	38.42
	80	36.72	43.44	**48.88**	**71.22**	**65.48**
Mixture at $P_{FA} = 5\%$	256	0.00	0.00	0.00	49.07	53.49
	80	**18.50**	**34.76**	**55.02**	**61.38**	**67.40**

tion of adversarial stego images effectively benefits the performance of CNN steganalyzers in the real-world scenario. Specifically, under payload 0.5 bpp (bit per pixel), the detection accuracy improvement is 19.95%.

4.4. Ablation Study

4.4.1. Statistical vectors

Statistical vectors are utilized to module the correlation between patches and universal statistical difference between cover and stego images. As shown in Table 3, the removal of statistical vectors decrease the detection accuracy of patch steganalysis on the mixture of adversarial and conventional stego images at 5% false alarm rate except under payload 0.2 bpp. Such experiment results indicate the statistical vectors effectively improve the detection ability of patch steganalysis.

4.4.2. Training image size

Table 5: The AUCs (area under the curve) of the patch steganalysis of different value of stride.

	Stride				
	8	16	24	32	48
AUC	0.9094	0.9038	0.9044	0.9091	**0.9143**

In this paper, we train the target model using 80 × 80 cropped image pairs to improve the detection accuracy on cover and conventional stego images. As shown in Table 4, the patch steganalysis trained on 256 × 256 image pairs obtains clearly lower detection accuracy on the mixture at 5% false alarm rate than that trained on 80×80 image pairs. While more robustness is obtained by the model trained on 256×256 images under relatively low payloads. Spatially, under payload 0.1 to 0.3 bpp, when the false alarm rate is 5%, the model trained on 256 × 256 images is predicting all the input images as cover. Only at the 10% false alarm rate, valid detection accuracies can be obtained, which are 20.34%, 34.57% and 45.33%.

The selection of the training image size can be considered a trade-off between robustness against adversarial steganography and detection accuracy on common samples. The experiments show that using 80 × 80 images to train patch steganalysis generates models of higher comprehensive detection ability. Thus, we set the model trained on $b \times b$ cropped images.

4.4.3. Sampling parameters

Table 6: The AUCs (area under the curve) of the patch steganalysis of different value of representative patch number.

	Patch number				
	2	5	10	15	20
AUC	0.8299	0.8822	0.9164	**0.9201**	0.9181

Table 7: The AUCs (area under the curve) of the patch steganalysis of different value of patch size.

	Patch size				
	48	64	80	96	128
AUC	0.8128	0.8399	**0.8502**	0.8453	0.8458

There are several parameters involved in patch sampling: stride, i.e., the distance between the starting pixels of neighboring patches, patch number and patch size. In this section, we compare the AUC (area under the curve) when cover : stego : adv = 2 : 1 : 1 to optimize the sampling parameters. Without loss the generality, only the results under payload 0.4 bpp are exhibited. Since training steganalyzers on datasets of different image sizes is quite time-consuming, the patch size parameter is optimized as the steganalyzer is trained on 256 × 256 images. The optimization of each parameter is conducted with the others fixed. The stride value is optimized with $g = 10$ and $b = 80$. The representative number (the group number) is optimized with $s = 24$ and $b = 80$. The patch size is optimized with $g = 10$ and $s = 24$. The statistics are shown in Table 5, Table 6 and Table 7. It is clear the optimal value of stride s, patch number (group number) g and patch size b are 48, 15 and 80 respectively.

5. CONCLUSION

Adversarial steganography severely challenges the security of CNN steganalyzers and their applications in reality. Previously, preprocessing is considered not feasible for steganalysis. In this paper, we propose a novel preprocessing based method, patch steganalysis, which utilizes random patch sampling and feature fusion to defend against adversarial steganography and minimize the defects on the detection of common samples. The experiment results show that the proposed method significantly improves the robustness of CNN steganalyzers and outperforms the previous works in real-world scenarios where there are adversarial stego images.

6. REFERENCES

[1] Jessica J. Fridrich and Tomás Filler, "Practical methods for minimizing embedding impact in steganography," in *Security, Steganography, and Watermarking of Multimedia Contents IX,*

San Jose, CA, USA, January 28, 2007, Edward J. Delp III and Ping Wah Wong, Eds. 2007, vol. 6505 of *SPIE Proceedings*, p. 650502, SPIE.

[2] Tomás Filler, Jan Judas, and Jessica J. Fridrich, “Minimizing additive distortion in steganography using syndrome-trellis codes,” *IEEE Trans. Inf. Forensics Secur.*, vol. 6, no. 3-2, pp. 920–935, 2011.

[3] Weixiang Li, Weiming Zhang, Li Li, Hang Zhou, and Nenghai Yu, “Designing near-optimal steganographic codes in practice based on polar codes,” *IEEE Trans. Commun.*, vol. 68, no. 7, pp. 3948–3962, 2020.

[4] Vojtech Holub and Jessica J. Fridrich, “Digital image steganography using universal distortion,” in *ACM Information Hiding and Multimedia Security Workshop, IH&MMSec ’13, Montpellier, France, June 17-19, 2013*, William Puech, Marc Chaumont, Jana Dittmann, and Patrizio Campisi, Eds. 2013, pp. 59–68, ACM.

[5] Bin Li, Ming Wang, Jiwu Huang, and Xiaolong Li, “A new cost function for spatial image steganography,” in *2014 IEEE International Conference on Image Processing, ICIP 2014, Paris, France, October 27-30, 2014*. 2014, pp. 4206–4210, IEEE.

[6] Linjie Guo, Jiangqun Ni, Wenkang Su, Chengpei Tang, and Yun-Qing Shi, “Using statistical image model for JPEG steganography: Uniform embedding revisited,” *IEEE Trans. Inf. Forensics Secur.*, vol. 10, no. 12, pp. 2669–2680, 2015.

[7] Jessica Fridrich and Jan Kodovsky, “Rich models for steganalysis of digital images,” *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 868–882, 2012.

[8] Xiaofeng Song, Fenlin Liu, Chunfang Yang, Xiangyang Luo, and Yi Zhang, “Steganalysis of adaptive JPEG steganography using 2d gabor filters,” in *Proceedings of the 3rd ACM Workshop on Information Hiding and Multimedia Security, IH&MMSec 2015, Portland, OR, USA, June 17 - 19, 2015*, Adnan M. Alattar, Jessica J. Fridrich, Ned M. Smith, and Pedro Comesaña Alfaro, Eds. 2015, pp. 15–23, ACM.

[9] Jan Kodovský, Jessica J. Fridrich, and Vojtech Holub, “Ensemble classifiers for steganalysis of digital media,” *IEEE Trans. Inf. Forensics Secur.*, vol. 7, no. 2, pp. 432–444, 2012.

[10] Rémi Cogranne, Vahid Sedighi, Jessica J. Fridrich, and Tomás Pevný, “Is ensemble classifier needed for steganalysis in high-dimensional feature spaces?,” in *2015 IEEE International Workshop on Information Forensics and Security, WIFS 2015, Roma, Italy, November 16-19, 2015*. 2015, pp. 1–6, IEEE.

[11] Jian Ye, Jiangqun Ni, and Yang Yi, “Deep learning hierarchical representations for image steganalysis,” *IEEE Trans. Inf. Forensics Secur.*, vol. 12, no. 11, pp. 2545–2557, 2017.

[12] Mehdi Boroumand, Mo Chen, and Jessica J. Fridrich, “Deep residual network for steganalysis of digital images,” *IEEE Trans. Inf. Forensics Secur.*, vol. 14, no. 5, pp. 1181–1193, 2019.

[13] Weike You, Hong Zhang, and Xianfeng Zhao, “A siamese CNN for image steganalysis,” *IEEE Trans. Inf. Forensics Secur.*, vol. 16, pp. 291–306, 2021.

[14] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus, “Intriguing properties of neural networks,” in *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun, Eds., 2014.

[15] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy, “Explaining and harnessing adversarial examples,” in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun, Eds., 2015.

[16] Nicholas Carlini and David A. Wagner, “Towards evaluating the robustness of neural networks,” in *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*. 2017, pp. 39–57, IEEE Computer Society.

[17] Yiwei Zhang, Weiming Zhang, Kejiang Chen, Jiayang Liu, Yujia Liu, and Nenghai Yu, “Adversarial examples against deep neural network based steganalysis,” in *Proceedings of the 6th ACM Workshop on Information Hiding and Multimedia Security, Innsbruck, Austria, June 20-22, 2018*, Rainer Böhme, Cecilia Pasquini, Giulia Boato, and Pascal Schöttle, Eds. 2018, pp. 67–72, ACM.

[18] Weixuan Tang, Bin Li, Shunquan Tan, Mauro Barni, and Jiwu Huang, “Cnn-based adversarial embedding for image steganography,” *IEEE Trans. Inf. Forensics Secur.*, vol. 14, no. 8, pp. 2074–2087, 2019.

[19] Chuan Guo, Mayank Rana, Moustapha Cissé, and Laurens van der Maaten, “Countering adversarial images using input transformations,” in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. 2018, OpenReview.net.

[20] Fangzhou Liao, Ming Liang, Yinpeng Dong, Tianyu Pang, Xiaolin Hu, and Jun Zhu, “Defense against adversarial attacks using high-level representation guided denoiser,” in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. 2018, pp. 1778–1787, Computer Vision Foundation / IEEE Computer Society.

[21] Swami Sankaranarayanan, Arpit Jain, Rama Chellappa, and Ser-Nam Lim, “Regularizing deep networks using efficient layerwise adversarial training,” in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, Sheila A. McIlraith and Kilian Q. Weinberger, Eds. 2018, pp. 4008–4015, AAAI Press.

[22] Solène Bernard, Patrick Bas, John Klein, and Tomás Pevný, “Explicit optimization of min max steganographic game,” *IEEE Trans. Inf. Forensics Secur.*, vol. 16, pp. 812–823, 2021.

[23] Guanshuo Xu, Han-Zhou Wu, and Yun-Qing Shi, “Structural design of convolutional neural networks for steganalysis,” *IEEE Signal Process. Lett.*, vol. 23, no. 5, pp. 708–712, 2016.

[24] Patrick Bas, Tomáš Filler, and Tomáš Pevnỳ, “” break our steganographic system”: the ins and outs of organizing boss,” in *International workshop on information hiding*. Springer, 2011, pp. 59–70.

[25] Patrick Bas and Teddy Furon, “BOWS-2 Contest (Break Our Watermarking System),” Organized between the 17th of July 2007 and the 17th of April 2008.