StegColNet: Steganalysis Based on an Ensemble Colorspace Approach

Shreyank N. Gowda[1(✉)] and Chun Yuan[2]

[1] University of Edinburgh, Edinburgh, UK
[2] Tsinghua University, Beijing, China

Abstract. Image steganography refers to the process of hiding information inside images. Steganalysis is the process of detecting a steganographic image. We introduce a steganalysis approach that uses an ensemble color space model to obtain a weighted concatenated feature activation map. The concatenated map helps to obtain certain features explicit to each color space. We use a levy-flight grey wolf optimization strategy to reduce the number of features selected in the map. We then use these features to classify the image into one of two classes: whether the given image has secret information stored or not. Extensive experiments have been done on a large scale dataset extracted from the Bossbase dataset. Also, we show that the model can be transferred to different datasets and perform extensive experiments on a mixture of datasets. Our results show that the proposed approach outperforms the recent state of the art deep learning steganalytical approaches by 2.32% on average for 0.2 bits per channel (bpc) and 1.87% on average for 0.4 bpc.

Keywords: Steganalysis · Color spaces · Greywolf optimization · Concatenated feature maps

1 Introduction

Steganography is a means of covert communication in which secret information is embedded into some form of digital media, such as an image, video or text file [3]. Usually, this form of embedding is done such that there is no apparent perceptible change in the embedding file. In multimedia security, steganography forms a critical research topic [4]. The difference between steganography and cryptography is that in cryptography data is encrypted and although difficult to break, raises a doubt in the mind of an attacker about the presence of secret information. Steganography, on the other hand, aims to reduce the risk of being detected. In general, images are considered as the embedding medium due to minute changes in an image being imperceptible to the human eye [4]. There are three main properties that a steganographic algorithm should possess: security, robustness, and capacity. In case of an image steganographic algorithm, security would mean how securely the algorithm can hide information, i.e., how little visual change is caused on an image using an image steganography algorithm.

A. Torsello et al. (Eds.): S+SSPR 2020, LNCS 12644, pp. 313–323, 2021.
https://doi.org/10.1007/978-3-030-73973-7_30

Robustness refers to the invariability of the steganographic algorithm when an image is subject of different transforms such as scaling, resizing, rotation, etc. The capacity for a steganographic algorithm represents the amount of data that can be embedded in an image before there is a noticeable visual change in the image [5]. Steganalysis is the process of detecting if a given image has information hidden in it or not [27]. In this regard, we can convert this problem into that of a simple classification problem. To detect if an image is embedded with information we propose the use of an ensemble color space model. Recently, it was seen an ensemble colorspace model [1] obtained excellent results on large scale image classification datasets such as imagenet [2]. Based on [1] we propose a novel steganalysis approach.

Steganalysis is the process of detecting if a given image has information hidden in it or not. In this regard, we can convert this problem into that of a simple classification problem. To detect if an image is embedded with information, we propose the use of an ensemble color space model.

We do the following:

- We use a colorspace approach to determine if an image is hiding information or not. We use ColorNet [1] and take the final activation map from each colorspace.
- We use weighted averaging to obtain a single feature map from all the individual feature maps that are generated by each colorspace. It was seen [1] that each color space had features explicit to themselves and this would help us detect minute changes in the image.
- We then use a levy-flight grey wolf optimization method (meta-heuristic approach) to select a smaller subset of features. Using these features, we classify the given image into one of two classes: containing concealed information or not.

1.1 Steganography

Most steganography algorithms can be expressed in Fig. 1. An image is broken down to it's RGB (Red Green Blue) channels and pixels in the individual channels are modulated with some cost function 'C' which embeds information into that channel. The most straightforward steganography algorithm is the LSB (Least Significant Bit) algorithm. Here, as the name suggests the least significant bit is taken, and one bit of information is stored (either as a 1 or a 0).

Steganography algorithms can be classified broadly into four categories: 1) cover image size 2) embedding domain-based algorithms 3) nature of retrieval based algorithms 4) adaptive steganographic algorithms. In the case of 2-D images, the information is embedded onto the 2-D plane of the cover image. This embedding can be done over transform domain coefficients (such as discrete cosine transforms, Fourier transforms, etc.) or on the spatial domain (an example is LSB). The 3-D approaches essentially follow the same general procedure. However, the procedure is repeated on multiple planes (for instance RGB

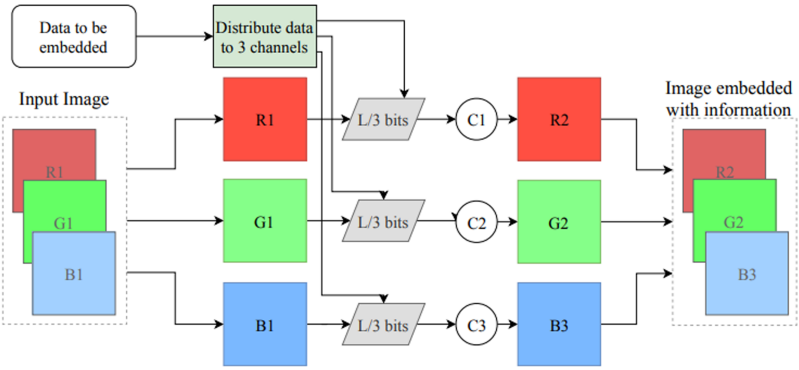

Fig. 1. Pipeline of a standard steganography algorithm (Color figure online)

in a color image has 3 planes that can embed information). Image steganography on 3-D images can be made in either geometrical domain [5], representation domain [6] or topological domain [7].

Some of the transform-based steganographic algorithms include discrete Fourier transform (DFT) [9], discrete cosine transform (DCT), discrete wavelet transform [10], complex wavelet transform [11] among others. Here, frequency coefficients obtained after applying transforms are used to hide secret bits. Along with the security being improved, these algorithms are robust to image compression, cropping, scaling, etc. Off late, machine learning approaches have been proposed such as SVM (Support Vector Machine)[12], genetic algorithm approaches [13], neural network-based steganography [14]. Though these approaches are black-box approaches, they have shown good results.

1.2 Steganalysis

Steganalysis is the method of trying to either determine a stego image (image where information is hidden) or extract the secret information. Our method deals with the former. We treat the problem at hand to be a classification problem, wherein, each image either contains some hidden information or not.

There are two basic approaches to steganalysis: signature steganalysis and statistical steganalysis. Signature steganalysis is the method wherein patterns, or signatures relevant to various steganographic algorithms are searched for. The presence of a pattern indicating that secret information is being hidden in the image. The quintessential process here is the repetition of patterns due to embedded secret information. The statistical approach searches for mathematical results to determine if the information is being hidden. Signature steganalysis is further classified into specific embedding [16] and universal blind steganalysis [15]. Specific embedding approaches are impractical because we need to know what steganography approach has been used to embed information. Hence, universal blind steganalysis [8,17] is preferred. These approaches help in the extraction of high dimensional features. However, the curse of dimensionality occurs.

Hence, a need to reduce feature size occurs. Some commonly used algorithms to do the same include wrappers, filters, etc. Filters are less complex; however, they perform poorly. Wrapper methods evaluate feature subset using predictive models [18]. However, wrappers are complex and time-consuming.

To overcome this, meta-heuristic approaches have been deployed. These approaches solve optimization problems by utilizing natural phenomena [19,20]. It was seen that Grey Wolf Optimization (GWO) performed better than other metaheuristic approaches for solving non-linear problems in a multi-dimensional space [19]. However, it has a slow convergence rate and gets trapped in local optima at times. It has been seen that GWO can be optimized by modifying it's parameter **A** to obtain a quick convergence rate, better convergence precision and higher agility for global searching.

2 Proposed Approach

2.1 Overall Architecture and Effect of Using Color Spaces

We consider steganalysis as a 2 class classification problem. The overall architecture is described in Fig. 2. The experimental analysis along with details regarding training set etc. are explained in the next section. Recently, the effect of color spaces on image classification has been explored [1]. It was seen that individual color spaces inherited classification features explicitly to themselves. This helped us ponder about the ability to extract information in an image where there is secret information being embedded. Colornet [1] being an ensemble model, that could extract features specific to each colorspace, was an excellent choice to utilize to help us in determining if an image could have information hidden in it. The output of Colornet is a high-dimensional vector, which causes a computationally intensive execution. To reduce the number of features selected we have to use an optimization approach for feature selection. Figure 1 shows the architecture of the model.

2.2 Optimization Process for Feature Selection

Feature Selection Using LF-Grey Wolf Optimization. In GWO, the head of the pack is the α. The next level of the hierarchy is β, δ and finally followed by ω. GWO models the social hierarchy and mathematically illustrates the hunting procedure as an optimization problem. If $\mathbf{X}_\mathrm{p}(\mathrm{t})$ and $\mathbf{X}(\mathrm{t})$ represent the position of prey and wolf at iteration 't', we can mathematically model the encircling process [19] with two coefficients **A** and **C** as shown in (1). **A** and **C** are calculated by (2).

$$\mathbf{D} = |\mathbf{C}.\mathbf{X}_p(t) - \mathbf{X}(t)|; \mathbf{X}(t+1) = \mathbf{X}_p(t) - \mathbf{A}.\mathbf{D} \tag{1}$$

$$\mathbf{A} = \mathbf{2a}.\mathbf{r_1} - \mathbf{a}; \mathbf{C} = \mathbf{2}.\mathbf{r_2} \tag{2}$$

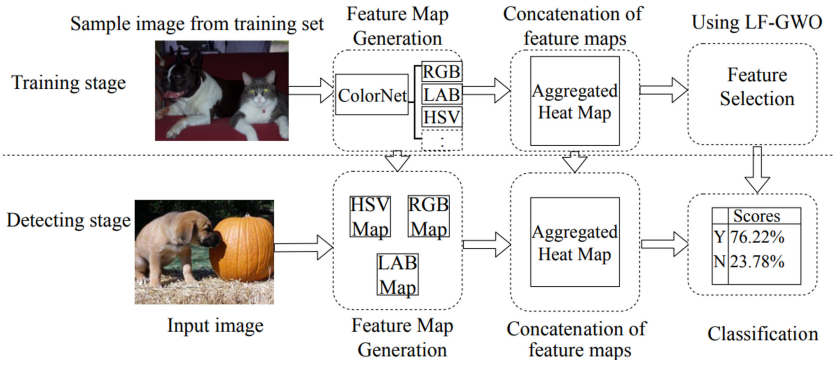

Fig. 2. Two phases involved in the overall architecture of the model: training the model using colornet and detecting stego-image using feature map aggregation

Here, $\mathbf{r}_1$ and $\mathbf{r}_2$ are random vectors in [0,1], **a** is a parameter that decreases linearly from 2 to 0 over iterations and also helps to control step size **D** of a grey wolf. Implementation of the end of the hunting process is done by decreasing the value of **A** which in turn depends on **a**. Once **a** turns zero, it means that the wolves have stopped moving. The linear decrease in **A** helps to exploit search space with minimal exploration. Hence, this traps a local optimum.

The size of the aggregated feature map creates an issue in terms of the complexity of the algorithm and the overall time needed for execution. To deal with this, we propose the use of levy flight-based grey wolf optimization (LF-GWO) for feature selection based on Levy probability function in (3). Here, μ represents position parameter, γ represents scale parameter and η represents the collection of samples in the distribution. The above equation holds good for all positive values of μ and 0 otherwise. The parameter **A** is modified by the Levy flight function as $\mathbf{A} = \mathbf{L(S)*r1}$. This makes **A** take up values in a non-linear decrease. **S** is the position of the wolf and **r1** is a random vector.

$$\mathbf{L}(\eta, \gamma, \mu) = \frac{\sqrt{\gamma}}{\mathbf{2}\pi}\mathbf{exp}[-\frac{\gamma}{\mathbf{2}(\eta-\mu)}]\frac{\mathbf{1}}{(\eta-\mu)^{\frac{\mathbf{3}}{\mathbf{2}}}} \tag{3}$$

The reason for selection of LF-GWO is based in the statistical results obtained in [21]. It was seen that for 15 defined benchmark functions, the wilcoxon rank sum test of LF-GWO outperforms existing optimization approaches in terms of mean fitness values. For further technical analysis please refer [21].

3 Experimental Analysis

3.1 Datasets and Training

Most commonly used steganalysis datasets are the Bossbase [22] and BOWS2 [23]. Each contains 10000 grayscale images. However, the approach proposed is

dependent on color, and as such, we use a dataset with color images. Hence, starting with the 10000 images of Bossbase [22] dataset, we generate a dataset by following the process done in [24]. We downsampled the full-resolution images to a size of 512×512. We then followed the process in [25], so that the training and testing scenarios were conducted in a similar environment. In [25], two datasets were created by using two demosaicing algorithms: Patterned pixel grouping (PPG) and Adaptive Homogeneity-Directed (AHD) and named BOSS-PPG-LAN and BOSS-AHD-LAN correspondingly. Further, by removing the down-sampling method, we can obtain two more datasets: BOSS-PPG-CRP and BOSS-AHD-CRP. By pairing a demosaicing algorithm with bilinear or bicubic kernels, we obtain four more datasets: BOSS-PPG-BIL, BOSS-AHD-BIL, BOSS-AHD-BIL, and BOSS-AHD-BIC.

We train our model by utilizing mini-batch stochastic gradient descent with the following parameters: learning rate: 0.0001, weight decay: 0.0005, step size: 5000, momentum: 0.75, gamma: 0.75, batch size: 32, maximum iterations: 40×10^4. Testing of the trained model was done for every 5000 iterations and accuracy in 40×104 iterations. HILL, SUNIWARD, CMD-C-SUNIWARD and CMD-C-HILL: 4 state of the art color steganography algorithms, were used as attacking targets for experimental analysis. The embedding payload was set to 0.2 bpc (bits per channel/band pixel) and 0.4 bpc. In order to select the most challenging scenarios and also follow similar conditions for result comparison, we followed the process executed in WISERNet [25].

3.2 Results Comparison

To compare our results, we considered three deep learning approaches for color steganalyzers, that are widely considered state of the art approaches: WISERNet [25], Deep Hierarchical Representations (DHR) [26] and Deep-CNN [27]. Experiments were conducted on the same datasets and using similar resources for a fair comparison. Popular steganography methods such as SUNIWARD [28], MiPOD [29], HILL [30] adopt an additive embedding distortion approach for minimizing framework [31]. Recently, CMD-C was proposed [32] by improvising the CMD approach for color images. We denote the CMD-C method using SUNIWARD and HILL as CMD-C-SUNIWARD and CMD-C-HILL respectively. Although DHR [26] and D-CNN [27] can be executed in channel-wise convolution, normal convolution and input concatenation as seen in [25], we show results only for the normal convolution as WiserNet [25] outperforms DHR and D-CNN in all cases. We also compare results with channel gradient correlation (CGC) [34].

The parameters used in terms of batch size and iterations were the same for all the comparisons. The other parameters were used as described in the original paper. Each experiment constituted 75% training images, i.e., 7500 images and 2500 images were used for testing. All experiments were performed 10 times and the average accuracy of testing was used. Table 1 compares the results of our approach with WISERNet (W-Net) [25], DHR [26], D-CNN [27], on BOSS-PPG-LAN (B-P-L), BOSS-PPG-BIC (B-P-Bc), BOSS-PPG-BIL (B-P-Bl), BOSS-AHD-BIC (B-A-Bc) and BOSS-AHD-BIL (B-A-Bl) with 0.2 bpc

and Table 2 with 0.4 bpc. As can be seen, the proposed method outperforms other state of the art methods for all but one case and also the percentage increase in detection is significant when patterned pixel grouping is performed on the datasets.

Table 1. Comparison of results for CMD-C-HILL stego images with 0.2 bpc. D-CNN is executed with 30 fixed SRM kernels. The best results are represented in bold font.

Dataset	DHR	D-CNN	W-Net	CGC	Proposed
B-P-L	0.6474	0.6562	0.7139	0.7231	**0.7741**
B-P-Bc	0.6589	0.7124	0.7318	0.7278	**0.7912**
B-P-Bl	0.7611	0.7487	0.8033	0.8120	**0.8316**
B-A-Bc	0.6614	0.6627	**0.7369**	0.7168	0.7368
B-A-Bl	0.7622	0.7647	0.8022	0.7981	**0.8044**

Table 2. Comparison of results for CMD-C-HILL stego images with 0.4 bpc. D-CNN is executed with 30 fixed SRM kernels. The best results are represented in bold font.

Dataset	DHR	D-CNN	W-Net	CGC	Proposed
B-P-L	0.7568	0.7941	0.8361	0.8268	**0.8724**
B-P-Bc	0.7732	0.8068	0.8435	0.8314	**0.8814**
B-P-Bl	0.87211	0.9045	0.9169	0.9165	**0.9381**
B-A-Bc	0.7728	0.8141	0.8448	0.8412	**0.8468**
B-A-Bl	0.8738	0.9067	**0.9144**	0.9044	0.9088

Further experimental analysis is done by mixing datasets as shown in [27]. Table 3 shows how the datasets were mixed. We further label the datasets in roman numerals for simplicity to display in the comparison of steganalyzers in Table 4 and 5. BPL, BPBc, BPBl, BABc, BABl, BAL are further abbreviations of BOSS-PPG-LAN, BOSS-PPG-BIC, BOSS-PPG-BIL, BOSS-AHD-BIC, BOSS-AHD-BIL and BOSS-AHD-LAN. Similarly to Tables 1 and 2, Table 4 compares results on the above-mentioned mixture of datasets with 0.2 bpc. Table 5 compares the results with 0.4 bpc. As can be seen, the proposed method outperforms recent state of the art approaches, by a significant margin.

Table 3. Representation of mixture of datasets. ✓ implies dataset has been selected and - implies otherwise.

Name	BPL	BPBc	BPBl	BABc	BABl	BAL
Set-I	✓	✓	✓	–	–	–
Set-II	–	–	–	✓	✓	✓
Set-III	✓	–	–	–	–	✓
Set-IV	✓	✓	✓	✓	✓	✓

Table 4. Comparison of results for CMD-C-HILL stego images with 0.2 bpc on mixture of datasets. D-CNN is executed with 30 fixed SRM kernels. The best results are represented in bold font.

Dataset	DHR	D-CNN	W-Net	CGC	PVC	Proposed
Set-I	0.7237	0.7259	0.7675	0.7712	0.7734	**0.8029**
Set-II	0.7214	0.7217	0.7714	0.7710	0.7684	**0.8026**
Set-III	0.6722	0.6865	0.7284	0.7412	0.7388	**0.7648**
Set-IV	0.7164	0.7182	0.7671	0.7782	0.7684	**0.8048**

Table 5. Comparison of results for CMD-C-HILL stego images with 0.4 bpc on mixture of datasets. D-CNN is executed with 30 fixed SRM kernels. The best results are represented in bold font.

Dataset	DHR	D-CNN	W-Net	CGC	PVC	Proposed
Set-I	0.8241	0.8289	0.8594	0.8788	0.8641	**0.9041**
Set-II	0.8231	0.8417	0.8806	0.8762	0.8661	**0.9021**
Set-III	0.7812	0.7892	0.8316	0.8411	0.8421	**0.8598**
Set-IV	0.8161	0.8214	0.8893	0.8796	0.8812	**0.9013**

4 Conclusion

With recent developments of color based steganography algorithms, the need for a powerful steganalyzer is needed. We saw recently, that an ensemble model of colorspaces has a significant impact on classification results. We propose StegColNet as a powerful color image steganalyzer. We employ an ensemble colorspace strategy to determine if an image is protecting information or not. We use ColorNet and take the final activation map from each colorspace. We use weighted averaging to obtain a single feature map from all the feature maps that are generated by each colorspace. We then use a levy-flight grey wolf optimization method to select a smaller subset of features. Using these features, we classify the given image into one of two classes: containing concealed information or not.

References

1. Gowda, S.N., Yuan, C.: ColorNet: investigating the importance of color spaces for image classification. In: Jawahar, C.V., Li, H., Mori, G., Schindler, K. (eds.) ACCV 2018. LNCS, vol. 11364, pp. 581–596. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-20870-7_36
2. Deng, J., et al.: ImageNet: a large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255. IEEE (2009)
3. Kahn, D.: The history of steganography. In: Anderson, R. (ed.) IH 1996. LNCS, vol. 1174, pp. 1–5. Springer, Heidelberg (1996). https://doi.org/10.1007/3-540-61996-8_27
4. Cheddad, A., Condell, J., Curran, K., Mc Kevitt, P.: Digital image steganography: survey and analysis of current methods. Signal Process. **90**(3), 727–752 (2010)
5. Li, N., Hu, J., Sun, R., Wang, S., Luo, Z.: A high-capacity 3D steganography algorithm with adjustable distortion. IEEE Access **5**, 24457–24466 (2017)
6. Tsai, Y.-Y.: An adaptive steganographic algorithm for 3D polygonal models using vertex decimation. Multimedia Tools Appl. **69**(3), 859–876 (2012). https://doi.org/10.1007/s11042-012-1135-8
7. Cheng, Y.M., Wang, C.M.: A high-capacity steganographic approach for 3D polygonal meshes. Visual Comput. **22**(9–11), 845–855 (2006)
8. Chakraborty, S., Jalal, A.S., Bhatnagar, C.: LSB based non blind predictive edge adaptive image steganography. Multimedia Tools Appl. **76**(6), 7973–7987 (2016). https://doi.org/10.1007/s11042-016-3449-4
9. Jayaram, P., Ranganatha, H.R., Anupama, H.S.: Information hiding using audio steganography-a survey. Int. J. Multimedia Appl. (IJMA) **3**, 86–96 (2011)
10. Kumar, V., Kumar, D.: A modified DWT-based image steganography technique. Multimedia Tools Appl. **77**(11), 13279–13308 (2017). https://doi.org/10.1007/s11042-017-4947-8
11. Narasimmalou, T., Joseph, R.A.: Discrete wavelet transform based steganography for transmitting images. In: IEEE-International Conference On Advances In Engineering, Science And Management (ICAESM-2012), pp. 370–375. IEEE (2012)
12. Gowda, S.N.: Innovative enhancement of the Caesar cipher algorithm for cryptography. In: 2016 2nd International Conference on Advances in Computing, Communication and Automation (ICACCA) (Fall), pp. 1–4. IEEE, September 2016
13. Chang, C.C., Yu, Y.H., Hu, Y.C.: Hiding secret data into an AMBTC-compressed image using genetic algorithm. In Second International Conference on Future Generation Communication and Networking Symposia, vol. 3, pp. 154–157. IEEE, December 2008
14. Gowda, S.N.: Using Blowfish encryption to enhance security feature of an image. In: 6th International Conference on Information Communication and Management (ICICM), pp. 126–129. IEEE, October 2016
15. Luo, X.Y., Wang, D.S., Wang, P., Liu, F.L.: A review on blind detection for image steganography. Signal Process. **88**(9), 2138–2157 (2008)
16. Fridrich, J., Goljan, M.: On estimation of secret message length in LSB steganography in spatial domain. In: Security, Steganography, and Watermarking of Multimedia Contents VI, vol. 5306, pp. 23–35. International Society for Optics and Photonics, June 2004
17. Kodovsky, J., Fridrich, J., Holub, V.: Ensemble classifiers for steganalysis of digital media. IEEE Trans. Inf. Forensics Secur. **7**(2), 432–444 (2012)

18. Deng, H. and Runger, G.: Feature selection via regularized trees. In: The 2012 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. IEEE, June 2012
19. Chhikara, R.R., Sharma, P., Singh, L.: An improved dynamic discrete firefly algorithm for blind image steganalysis. Int. J. Mach. Learn. Cybern. **9**(5), 821–835 (2016). https://doi.org/10.1007/s13042-016-0610-3
20. Yao, X., Liu, Y., Lin, G.: Evolutionary programming made faster. IEEE Trans. Evol. Comput. **3**(2), 82–102 (1999)
21. Pathak, Y., Arya, K.V., Tiwari, S.: Feature selection for image steganalysis using levy flight-based grey wolf optimization. Multimedia Tools Appl. **78**(2), 1473–1494 (2018). https://doi.org/10.1007/s11042-018-6155-6
22. Gowda, S.N.: Human activity recognition using combinatorial deep belief networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 1–6. (2017)
23. Bas, P., Filler, T., Pevný, T.: "Break our steganographic system": the ins and outs of organizing BOSS. In: Filler, T., Pevný, T., Craver, S., Ker, A. (eds.) IH 2011. LNCS, vol. 6958, pp. 59–70. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-24178-9_5
24. Piva, A. and Barni, M.: The first BOWS contest: break our watermarking system. In: Security, Steganography, and Watermarking of Multimedia Contents IX, vol. 6505, p. 650516. International Society for Optics and Photonics, February 2007
25. Goljan, M., Fridrich, J., Cogranne, R.: Rich model for steganalysis of color images. In: IEEE International Workshop on Information Forensics and Security (WIFS), pp. 185–190. IEEE, December 2014
26. Zeng, J., Tan, S., Liu, G., Li, B., Huang, J.: WISERNet: wider separate-then-reunion network for steganalysis of color images. IEEE Trans. Inf. Forensics Secur. **14**(10), 2735–2748 (2019)
27. Ye, J., Ni, J., Yi, Y.: Deep learning hierarchical representations for image steganalysis. IEEE Trans. Inf. Forensics Secur. **12**(11), 2545–2557 (2017)
28. Xu, G.: Deep convolutional neural network to detect J-UNIWARD. In: Proceedings of the 5th ACM Workshop on Information Hiding and Multimedia Security, pp. 67–73. ACM, June 2017
29. 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
30. Sedighi, V., Cogranne, R., Fridrich, J.: Content-adaptive steganography by minimizing statistical detectability. IEEE Trans. Inf. Forensics Secur. **11**(2), 221–234 (2016)
31. Li, B., Wang, M., Huang, J., Li, X.: A new cost function for spatial image steganography. In: IEEE International Conference on Image Processing (ICIP), pp. 4206–4210. IEEE, October 2014
32. Fridrich, J., Filler, T.: Practical methods for minimizing embedding impact in steganography. In: Security, Steganography, and Watermarking of Multimedia Contents IX, vol. 6505, p. 650502. International Society for Optics and Photonics, February 2007
33. Tang, W., Li, B., Luo, W., Huang, J.: Clustering steganographic modification directions for color components. IEEE Signal Process. Lett. **23**(2), 197–201 (2016)
34. Qin, X., Li, B., Tan, S., Zeng, J.: A novel steganography for spatial color images based on pixel vector cost. IEEE Access **7**, 8834–8846 (2019)

35. Kang, Y., Liu, F., Yang, C., Xiang, L., Luo, X., Wang, P.: Color image steganalysis based on channel gradient correlation. Int. J. Distrib. Sensor Netw. **15**(5), 1550147719852031 (2019)
36. Gowda, S.N., Yuan, C.: Using an ensemble color space model to tackle adversarial examples (2020). arXiv preprint arXiv:2003.05005
37. Gowda, S.N.: An intelligent fibonacci approach to image steganography. In: 2017 IEEE Region 10 Symposium (TENSYMP), pp. 1–4. IEEE, July 2017