A New Steganalysis Method Using Densely Connected ConvNets

Brijesh Singh[1](✉), Prasen Kumar Sharma[1], Rupal Saxena[2], Arijit Sur[1], and Pinaki Mitra[1]

[1] Department of Computer Science and Engineering,
Indian Institute of Technology Guwahati, Guwahati, India
{brijesh.singh,kumar176101005,arijit,pinaki}@iitg.ac.in
[2] Department of Chemistry, Indian Institute of Technology Guwahati,
Guwahati, India
rupal.saxena@iitg.ac.in

Abstract. Steganography is an ancient art of communicating a secret message through an innocent-looking image. On the other hand, steganalysis is the counter process of the steganography, which targets to detect hidden trace within a given image. In this paper, a new approach to steganalysis is presented to learn prominent features and avoid loss of stego signals. The proposed model uses diverse sized filters to capture all useful steganalytic features through a densely connected convolutional network. Moreover, there is no fully connected network in the proposed model, which allows testing any size of images regardless of the image size used for training. To justify the applicability of the proposed scheme, it has been shown experimentally that the proposed scheme outperforms most of the related state-of-the-art methods.

Keywords: Steganography · Steganalysis · Convolutional Neural Network · DenseNet

1 Introduction

Steganography is the process of concealing secret information within an ordinary image. The image which is used for hiding the secret information is called a *cover* image. The image after embedding a secret message is known as a *stego* image. Steganography can be categorized into two types- (1) Spatial domain and (2) Transform domain steganography. The spatial domain steganography hides the secret by modifying the pixels of the cover image. The DCT domain steganography conceals the secret message within the DCT coefficients [1] of the image. One of the traditional steganography schemes such as LSB replacement [20] hides the secret information in the least significant bits (LSB) of the image pixels. Modern steganography schemes such as HUGO [15], wow [5], S-UNIWARD [6] hides a secret message by minimizing some heuristically defined distortion function. The distortion function assigns a high cost when embedding in the smooth regions in the image whereas a low cost to the noisy areas of the image.

B. Deka et al. (Eds.): PReMI 2019, LNCS 11941, pp. 277–285, 2019.
https://doi.org/10.1007/978-3-030-34869-4_31

Steganalysis is the process of detecting the trace of the hidden message in the given image. Steganalysis can be broadly divided into two types: (1) Blind and (2) Targeted steganalysis. *Blind steganalysis* detects the embedding in the image without knowing the steganographic algorithm used for embedding. The *Targated steganalysis* utilizes the knowledge of the steganographic scheme used for embedding. Steganalysis methods work in two stages. In the first stage, features are extracted using some tools, and in the second stage, classification is done based on the extracted features. The distortion function based recent steganographic schemes are more likely to distribute the secret message in the noisy or high textured area of the image than the flat areas. Therefore, the steganalysis schemes assume that the steganographic noise lies in the high-frequency components of the source image, thereby, strive to capture these feature for steganalysis.

2 Related Works

A lot of steganalysis works have been reported in the literature. These works can be broadly categorized into two types: (1) Handcrafted feature based and (2) Deep feature based steganalysis.

The conventional handcrafted feature based methods use some fixed handcrafted filters to extract the steganalytic features which are used for steganalytic classification. Pevny et al. introduced the Subtractive Pixel Adjacency Matrix (SPAM) [14], which utilized the fact that the steganographic noise alters the dependencies between the neighboring pixels of an image. Using higher-order Markov chain [19] these dependencies are captured. The transition probability matrix of the Markov chain is used as features to train an SVM classifier [4] for steganalysis. The Spatial Rich Model (SRM) [3] is proposed by Fridrich and Kodovsky, which uses several linear and non-linear filters to compute noise residual, followed by 106 different submodels to capture diverse kinds of relationships between neighboring pixels of noise residual. The submodels are used to train the Ensemble Classifier (EC) [10] for steganalytic classification. SRM [3] showed considerable improvement in detecting the trace of steganographic embedding in images over SPAM [14]. The performance of classifiers depends on the quality of features supplied to the classifiers. Handcrafted feature based schemes such as SRM [3] and SPAM [14] which rely on several fixed handcrafted filters, may be suboptimal in extracting all the precise steganalytic features.

Convolutional Neural Networks (CNN) are known for best automatic feature extractors which mitigate the problems of handcrafted feature extraction. Recently, many steganalysis works have been reported in the literature; some of them are as follows: Qian et al. proposed GNCNN [16], a CNN based model for steganalysis. The GNCNN [16] comprise of a fixed preprocessing layer and five convolution layers for feature extraction, followed by three fully connected layers for classification. GNCNN used a Gaussian activation to capture stego and cover signals more precisely. The preprocessing layer has a fixed high-pass filter, which exposes the stego noise and suppresses the image component. GNCNN reported a comparable results with SRM [3] on S-UNIWARD [6], HUGO [15], and WOW [5]. Xu et al. proposed XuNet [21] comprised of a preprocessing layer with a high-pass filter and five groups of layers for feature extraction followed by a fully-connected network for classification. Each group consists of a convolution layer followed by an average pooling and Batch Normalization (BN) [8]. The first group used an ABS layer to capture all the values of noise residual (negative as well as positive values) which might be discarded by some of the activation functions, followed by a convolution layer. Authors claimed a considerable performance over SRM with EC when detecting HILL [11] and S-UNIWARD [6]. Ye et al. [22] proposed a CNN based framework which initializes the first layer of the model with the filters of SRM [3] to better capture the noise residual. They also introduced an activation function named truncated linear unit to capture noise residual with low SNR. Authors reported better performance as compared to SRM [3] for WOW [5], S-UNIWARD [6] and HILL [11] embedding. Tian and Li [18] proposed a CNN based steganalysis using transfer learning. The model used a Gaussian high-pass filter for the preprocessing of images followed by the pre-trained Inception-V3 [17] model for steganalytic classification.

It has been observed from the literature that most of the existing CNN based steganalysis schemes: (i) Sharply increase the feature space by using a sequence of kernels in subsequent layers. (ii) Use some fixed-size kernels (with less variation) which may not be much expressive in learning the stego features since the stego signal is weak and sparse in nature. A kernel with lower spatial dimension may not learn, and a kernel with higher spatial dimension may lead to overfitting. (iii) Use a fully connected layer at the end for classification. The use of fully-connected layers imposes a constraint that the training and testing must be carried out on the images with the same spatial dimension. In order to use images of different sizes, due to the restriction mentioned above, the images must be resized before testing. However, resizing may lead to loss of stego signals, conceptually similar to pooling.

In this paper, considering the shortcomings mentioned above, a densely connected convolution network for steganalysis has been proposed for steganalysis. In contrast to the existing schemes, the proposed scheme makes the following contributions:

- A densely connected convolutional network without pooling layers is proposed, which progressively captures the steganalytic features at different scales.
- The fully connected layers are removed, which allows the model to be tested on any size of images regardless of the size of images used for training.

The proposed scheme is trained and tested on BOSSBase 1.0 [2] dataset and the steganalytic performance is compared with SRM [3], SPAM [14] against S-UNIWARD [6], HUGO [15], WOW [5] and HILL [11]. The performance of the proposed scheme is also compared with a recently proposed scheme of Tian and Li [18] against WOW [5] and S-UNIWARD [6].

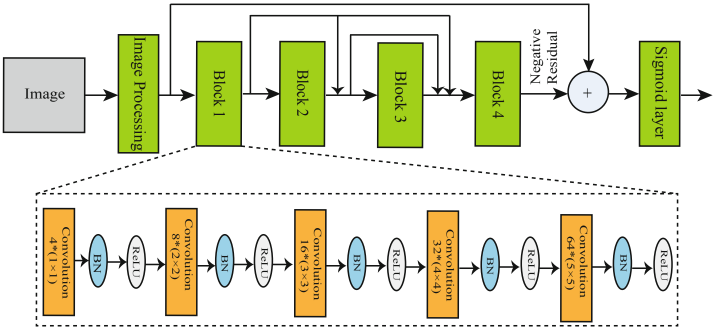

Fig. 1. The proposed model architecture. The architecture of each block is similar; one of the blocks (Block 1) is also shown in dotted box. Block consists of 4 × ($Conv \rightarrow BN \rightarrow ReLu$) with sizes indicated for each convolution block.

3 Proposed Work

This section presents the proposed scheme for *targeted* steganalysis. The proposed model is inspired by DenseNet [7]. The model architecture of the proposed scheme is shown in Fig. 1. The proposed model comprises of an image processing layer followed by four densely connected convolution blocks and a sigmoid layer at the end for classification. Since the steganalytic classifiers are trained on the noise residual instead of the image components, a fixed high-pass filter (HPF) given in Eq. (1) has been used in the image processing layer.

$$HPF = \frac{1}{12}\begin{pmatrix} -1 & 2 & -2 & 2 & -1 \\ 2 & -6 & 8 & -6 & 2 \\ -2 & 8 & -12 & 8 & -2 \\ 2 & -6 & 8 & -6 & 2 \\ -1 & 2 & -2 & 2 & -1 \end{pmatrix} \tag{1}$$

The kernel of the image processing layer is kept fixed and is not updated while training. The noise residual extracted from the image processing layer is used as input to the subsequent dense blocks. The densely connected blocks are used to avoid the problem of vanishing gradients and stego features. Each block is connected to all its subsequent blocks. Consequently, all the blocks receive the feature map from all their preceding blocks. Each block comprises of five convolutional layers. The details of the layers used in each block are given in Table 1. All the blocks have the same configuration except for the last block (Block 4), where the output feature size is 1 × 512 × 512. Convolutional layers in each block are followed by the Batch Normalization [8] for faster convergence and the ReLU [12] activation. Pooling layer has not been used since the use of pooling may result in loss of the stego noise. The number of convolutional filters progressively increase as $4, 8, 16, 32$ and 64, and the kernel size also increases gradually from 1 × 1 to 5 × 5 as each block slowly increases the scope of the convolution operator. The different sized kernels help to learn the features at different scales,

Table 1. Details of the layers in each dense block

Layer #	Input feature size	# of filters	Filter size	Output feature size
1	$1 \times 512 \times 512$	4	1×1	$4 \times 512 \times 512$
2	$4 \times 512 \times 512$	8	2×2	$8 \times 512 \times 512$
3	$8 \times 512 \times 512$	16	3×3	$16 \times 512 \times 512$
4	$16 \times 512 \times 512$	32	4×4	$32 \times 512 \times 512$
5	$32 \times 512 \times 512$	64	5×5	$64 \times 512 \times 512$

thereby avoiding the loss of stego signal and capturing more prominent features. The output of the densely connected blocks is a negative residual map which is pixel-wise added to noise residual extracted by the image processing layer to boost the noise components. The resulting output is used as input to the classification layer (sigmoid layer). The classification layer determines whether the input image is a stego or cover image by using the mean sigmoid over entire pixels. The whole framework is trained by minimizing the cross-entropy loss given in Eq. (2).

$$L = -\sum_{\forall x} p(x).\log(q(x)) \tag{2}$$

where $p(x)$ and $q(x)$ denotes the true and estimated distributions respectively, over a discrete variable x.

4 Implementation Details and Results

4.1 Experimental Setup

The experiments are carried out on BOSSBase v1.0 dataset [2]. The dataset consists of 10,000 cover images of size 512×512. The steganographic embedding algorithms[1] S-UNIWARD [6], HUGO [15], WOW [5] and HILL [11] are used to obtain stego images. Further, 10000 cover-stego pairs of images are divided into training: 5000, validation: 1000 and testing: 4000 cover-stego pairs. To compare the performance with the proposed model SRM [3] and SPAM [14] are implemented along with Ensemble Classifier v.2.0 [10], with the same split (5000 pairs for training and 5000 pairs for testing) as for proposed model. The proposed model is trained using Pytorch [13] on a standard workstation having NVIDIA Quadro M-4000 GPU (8 GB) for 90 epochs. The learning rate is initially set to 0.001 and decays by a factor of 10 every 30 epochs. The batch size is empirically kept as 8 (4 cover and 4 stego). Adam Optimizer [9] is used to optimize the proposed network parameters when training.

[1] Steganographic algorithms, feature extractors such as SRM, SPAM and Ensemble classifier can be found at: http://dde.binghamton.edu/download/.

Table 2. Steganalytic classification accuracy (in %) of the proposed scheme is compared to SRM [3] with Emsemble classifier [10] and SPAM [14] with Ensemble classifier against S-UNIWARD [6], HUGO [15], WOW [5] and HILL [11].

Scheme	Payload (bpp)	Proposed scheme	SRM with EC	SPAM with EC
S-UNIWARD	0.1	66.50%	59.05%	54.24%
	0.2	68.50%	62.17%	58.92%
	0.3	70.75%	65.80%	63.44%
	0.4	75.25%	73.70%	67.51%
HUGO	0.1	63.50%	60.03%	52.36%
	0.2	70.25%	67.98%	56.00%
	0.3	74.00%	74.46%	60.03%
	0.4	77.25%	78.30%	63.98%
WOW	0.1	67.25%	60.97%	52.46%
	0.2	70.75%	65.77%	55.82%
	0.3	74.00%	69.26%	58.98%
	0.4	76.50%	75.12%	62.33%
HILL	0.1	61.50%	55.45%	52.28%
	0.2	65.75%	61.37%	55.26%
	0.3	69.50%	67.11%	58.41%
	0.4	75.00%	72.58%	61.37%

Table 3. Comparison of the proposed scheme with Tian and Li [18] in terms of steganalytic classification accuracy (in %) against WOW [5] and S-UNIWARD [6].

Payload	WOW		S-UNIWARD	
bpp	Proposed scheme	Tian and Li [18]	Proposed	Tian and Li [18]
0.1	67.25%	67.90%	66.50%	65.10%
0.3	74.00%	69.00%	70.75%	67.20%
0.4	76.50%	71.4%	75.25%	69.80%

4.2 Results

The quantitative results for the proposed model are given in Table 2 when compared to the SRM with EC [3] and SPAM [14] with EC [10] against S-UNIWARD [6], HUGO [15], WOW [5], and HILL [11] steganographic schemes with different embedding rates. The results are measured in terms of percentage (%) classification accuracy. The best result is shown in the red color, and the blue color represents the second best result. A series of graphs are also given in Fig. 2 for a visual presentation where the proposed scheme is shown in red color, SRM with EC [3] is shown in green color and SPAM with EC [14] is shown in blue color. Results are evident that the proposed scheme outperformed SRM [3]

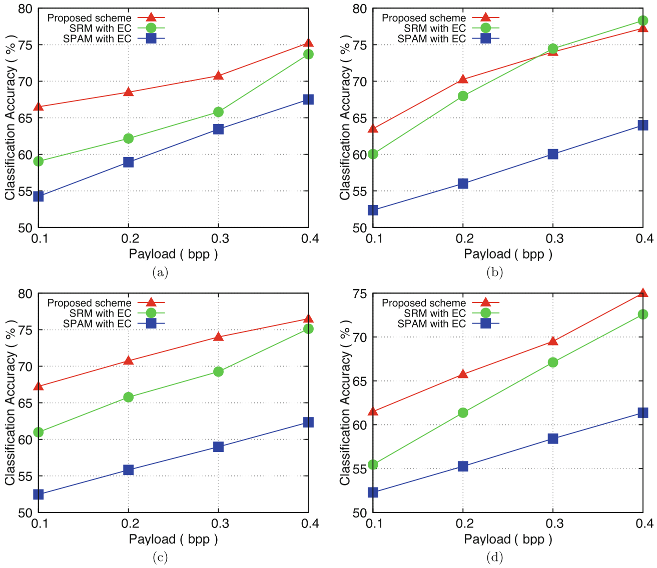

Fig. 2. Steganalytic performance comparison of the proposed scheme (Red) with SRM with EC (Green) and SPAM with EC (Blue) against: (a) S-UNIWARD (b) HUGO (c) WOW and (d) HILL steganography on embedding rates - {0.1, 0.2, 0.3, 0.4} bpp (Color figure online)

as well as SPAM [14] for most of the steganographic algorithms. The steganalytic performance of the proposed scheme is also compared with a recent work by Tian and Li [18], which has the same experimental setup in their work as the proposed scheme. The comparison is done against WOW [5] and S-UNIWARD [6] on embedding rates - {0.1, 0.3, 04} bits per pixel (bpp). The results are given in Table 3, the best result is shown in red color, and the next best is shown in blue color. The proposed scheme has comparable performance against WOW [5] on 0.1 bpp, and for the rest of steganographic embedding and payloads, the proposed scheme clearly outperformed Tian and Li [18].

5 Conclusion

In this paper, a densely connected convolution network based steganalysis is presented. The proposed model captures complex dependencies that are more appropriate for steganalysis, and the learned features avoid the loss of stego signals. The proposed model has no fully connected layer which adds advantage that the model can be tested on any size of the image unlike with fully-connected layers where the image size used for training and testing must be same. The steganalytic performance of the proposed scheme is compared with SRM, SPAM with Ensemble Classifier and a recent scheme by Tian and Li against different steganographic algorithm on different embedding rates. The proposed model outperforms the existing schemes with a considerable margin.

References

1. Ahmed, N., Natarajan, T., Rao, K.R.: Discrete cosine transform. IEEE Trans. Comput. **100**(1), 90–93 (1974)
2. 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
3. Fridrich, J., Kodovsky, J.: Rich models for steganalysis of digital images. IEEE Trans. Inf. Forensics Secur. **7**(3), 868–882 (2012)
4. Hearst, M.A., Dumais, S.T., Osuna, E., Platt, J., Scholkopf, B.: Support vector machines. IEEE Intell. Syst. Their Appl. **13**(4), 18–28 (1998)
5. Holub, V., Fridrich, J.: Designing steganographic distortion using directional filters. In: 2012 IEEE International Workshop on Information Forensics and Security (WIFS), pp. 234–239. IEEE (2012)
6. Holub, V., Fridrich, J., Denemark, T.: Universal distortion function for steganography in an arbitrary domain. EURASIP J. Inf. Secur. **2014**(1), 1 (2014)
7. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4700–4708 (2017)
8. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015)
9. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
10. Kodovsky, J., Fridrich, J., Holub, V.: Ensemble classifiers for steganalysis of digital media. IEEE Trans. Inf. Forensics Secur. **7**(2), 432–444 (2011)
11. Li, B., Wang, M., Huang, J., Li, X.: A new cost function for spatial image steganography. In: 2014 IEEE International Conference on Image Processing (ICIP), pp. 4206–4210. IEEE (2014)
12. Nair, V., Hinton, G.E.: Rectified linear units improve restricted Boltzmann machines. In: Proceedings of the 27th International Conference on Machine Learning (ICML 2010), pp. 807–814 (2010)
13. Paszke, A., Gross, S., Chintala, S., Chanan, G.: Pytorch: tensors and dynamic neural networks in Python with strong GPU acceleration. PyTorch: tensors and dynamic neural networks in Python with strong GPU acceleration 6 (2017)
14. Pevny, T., Bas, P., Fridrich, J.: Steganalysis by subtractive pixel adjacency matrix. IEEE Trans. Inf. Forensics Secur. **5**(2), 215–224 (2010)
15. Pevný, T., Filler, T., Bas, P.: Using high-dimensional image models to perform highly undetectable steganography. In: Böhme, R., Fong, P.W.L., Safavi-Naini, R. (eds.) IH 2010. LNCS, vol. 6387, pp. 161–177. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-16435-4_13
16. Qian, Y., Dong, J., Wang, W., Tan, T.: Deep learning for steganalysis via convolutional neural networks. In: Media Watermarking, Security, and Forensics 2015, vol. 9409, p. 94090J. International Society for Optics and Photonics (2015)
17. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818–2826 (2016)
18. Tian, J., Li, Y.: Convolutional neural networks for steganalysis via transfer learning. Int. J. Pattern Recognit Artif Intell. **33**(02), 1959006 (2019)

19. Tijms, H.C., Tijms, H.C.: Stochastic Models: An Algorithmic Approach, vol. 994. Wiley, Chichester (1994)
20. Wu, H.C., Wu, N.I., Tsai, C.S., Hwang, M.S.: Image steganographic scheme based on pixel-value differencing and LSB replacement methods. IEEE Proc.-Vis. Image Signal Process. **152**(5), 611–615 (2005)
21. 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)
22. Ye, J., Ni, J., Yi, Y.: Deep learning hierarchical representations for image steganalysis. IEEE Trans. Inf. Forensics Secur. **12**(11), 2545–2557 (2017)