Learning Representations for Steganalysis from Regularized CNN Model with Auxiliary Tasks

Yinlong Qian, Jing Dong, Wei Wang and Tieniu Tan

Abstract The key challenge of steganalysis is to construct effective feature representations. Traditional steganalysis systems rely on hand-designed feature extractors. Recently, some efforts have been put toward learning representations automatically using deep models. In this paper, we propose a new CNN based framework for steganalysis based on the concept of incorporating prior knowledge from auxiliary tasks via transfer learning to regularize the CNN model for learning better representations. The auxiliary tasks are generated by computing features that capture global image statistics which are hard to be seized by the CNN network structure. By detecting representative modern embedding methods, we demonstrate that the proposed method is effective in improving the feature learning in CNN models.

1 Introduction

The field of image steganalysis aims to reveal the presence of secret messages in digital images. It is often seemed as a pattern recognition problem. The key challenge of image steganalysis lies in building effective feature representations that are sensitive to stego signal while insensitive to image content. In order to obtain an accurate detection, it is important that the feature representations consider complex dependencies among individual image elements to capture the traces caused by embedding

Y. Qian
Department of Automation, University of Science and Technology of China, Hefei, China
e-mail: ylqian@mail.ustc.edu.cn

J. Dong (✉) · W. Wang · T. Tan
Center for Research on Intelligent Perception and Computing, Institute of Automation,
Chinese Academy of Sciences, Beijing, China
e-mail: jdong@nlpr.ia.ac.cn

W. Wang
e-mail: wwang@nlpr.ia.ac.cn

T. Tan
e-mail: tnt@nlpr.ia.ac.cn

Q. Liang et al. (eds.), *Proceedings of the 2015 International Conference on Communications, Signal Processing, and Systems*, Lecture Notes in Electrical Engineering 386, DOI 10.1007/978-3-662-49831-6_64

operations. In the past years, researchers have focus on designing appropriate feature extractors, and various features have been constructed to capture different types of dependencies [4–6, 10, 18, 21–23]. A classifier is then trained based on these features to distinguish between cover and stego images. Though significant progress has been achieved, the detection accuracy is far from satisfactory, especially when against new and advanced steganographic methods. Moreover, these traditional steganalysis methods are heavily dependent on expert experiences, and it is difficult and time-consuming to design new features.

More recently, many efforts have been put toward learning feature representations automatically for steganalysis using deep learning models, which are powerful in learning complex representations by transforming the inputs through multiple layers of nonlinear processing. For example, Qian et al. [19] propose a novel framework for steganalysis based on Convolutional Neural Network (CNN). In the proposed framework, both the feature extraction and classification stages are unified under a single architecture, and are trained simultaneously. One obvious advantage of such method is that it would greatly reduce the amount of human labor by leaving the design of the feature extractor to the learning algorithm. Another is that the end-to-end training make the model possible to automatically discover useful information directly from data, while exploiting the guidance of classification.

Inspired by the recent progress on feature learning for steganalysis, this work takes a new CNN based approach, in which incorporation of prior knowledge to regularize the learning process is considered to boost the performance of steganalysis. In fact, though deep learning models have shown great promise in learning powerful features for pattern recognition, the difficulty of training thousands of or even millions of parameters still exists. They are easy to over-fitting and getting stuck in local minima, especially when trained on small datasets. To reduce these problems during training and to improve the performance of the model, many regularization methods have been developed, such as dropout [7], dropconnect [24], stochastic pooling [25], and data augmentation [3]. In this paper, we propose regularizing the CNN models for steganalysis by encoding prior knowledge via transfer learning from auxiliary tasks. The auxiliary tasks are generated by computing features that capture global statistics which are hard to be seized by the CNN network structure. We expect to encode such information in to the model, and encourage the learned feature representations to capture the global statistics for better detection performance.

2 Related Work

Our method is related to numerous works on deep learning, feature based steganalysis, and transfer learning. In this section, we briefly discuss them below.

Deep learning: Deep learning is a class of machine learning methods that addresses the problem of what makes better representations and how to learn them. The deep learning models have deep architectures that consist of multiple levels of non-linear processing and can be trained to hierarchically learn complex representations by combining information from lower layers. There are many different types of deep models, such as Deep Boltzmann Machines [20], deep autoencoders [13], and Convolutional Neural Networks [14]. They have practically proved to be more powerful learning schemes for many artificial intelligence (AI) tasks such as object recognition, natural language processing, and image classification. In this paper, we focus on CNN as a base learner for steganalysis tasks. In a CNN model, trainable filters and pooling operations are applied alternatingly to the inputs, resulting in increasingly complex feature representations.

Feature based steganalysis: Most of the recent feature extraction methods for steganalysis follow a well-established paradigm of assembling a complex model as a combination of many diverse submodels to capture various dependencies among image elements [4–6, 10, 22, 23]. The submodels are constructed by firstly forming various noise residuals from pixels or DCT coefficients using a large number of designed linear or non-linear filters, and then computing global statistics such as high order co-occurrences from the residuals. Such methods rely heavily on expert human experiences to design different submodels to capture complementary information. Moreover, since the feature extraction and classification stages are independent, the guidance of classification can not be utilized for feature extraction. By contrast, our CNN based methods automatically learn features from data by training parameters in both feature extraction and classification stages. In [19], the authors also propose a CNN based model for feature learning in steganalysis. However,our method differs in that we exploit priori knowledge from auxiliary tasks to facilitate steganalysis feature learning for better performance.

Transfer learning: Transfer learning aims to leverage shared domain-specific knowledge contained in related tasks to help improving the learning of the target task. There has been a large amount of algorithms and techniques proposed on transfer learning to solve different problems. In this work, we mainly focus on transfer learning in the neural network. In [17], transfer learning with CNNs is explored for object recognition in a manner of reusing layers supervised trained on a large dataset to compute mid-level image representation for another dataset with limited training data. Differently to this work, we here transfer knowledge from some hand-engineered features. Similar ideas can also be found in other tasks [1, 11, 15], but its performance in steganalysis is not clear.

3 Proposed Framework

In this section, we will introduce the proposed framework in detail.

3.1 Exploiting Knowledge from Auxiliary Tasks with CNN

In our work, we use CNN, one of the most popular deep models, as the base learner to learn feature representations for steganalysis. A typical CNN model includes multiple convolutional layers, several fully-connected layers. The final layer is connected to a classifier for classification. In a convolutional layer, trainable filters, non-linearity and local pooling of feature maps using a max or an averaging operation are applied in sequence. Detailed descriptions of CNNs can be found in [14].

Though CNN has been proved to be a powerful learning tool, the training of a large CNN architecture is still a challenging task, especially when facing limited training data. The key idea of this work is that the priori knowledge provided from auxiliary tasks through transfer learning can help CNN learn better feature representations for steganalysis as illustrated in Fig. 1.

In the proposed framework, the CNN model described in Sect. 3.2 extracts feature representations with multiple convolutional layers, and passes them to several fully-connected layers. The outputs of the last fully connected layer are fed to a two-way softmax for classification task, which is the target task in this work. The loss function here is the cross-entropy loss, which we call the target loss.

Meanwhile, the constructed auxiliary tasks from input described in Sect. 3.3 are used to regularize the CNN model. This is achieved by connecting the output units of the auxiliary tasks to the last fully-connected layer of the CNN structure and computing the least square loss, which we called the auxiliary loss. In this case, it encourages the fully-connected layer information to be close to the information provided by auxiliary tasks.

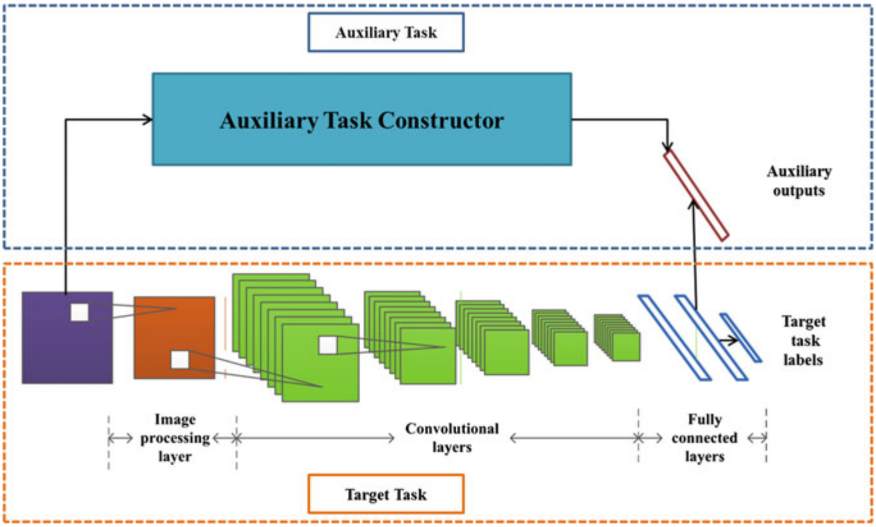

Fig. 1 Proposed framework for steganalysis

The overall loss function for the whole regularized model is a weighted summation of the target loss and the auxiliary loss. The network is then trained using backpropagation algorithm to minimize the overall loss, and thus the learning of features for steganalysis is guided by both the labeled information from the classification task and information from the auxiliary tasks simultaneously.

3.2 CNN Architecture

In this section, we describe the CNN architecture that is used for learning features for steganalysis as shown in Fig. 1. The architecture is composed of one image processing layer, five convolutional layers and three fully connected layers. It accepts an image patch of size 256 × 256 as input. Then the image processing layer computes the residuals with a predefined filter kernel of size 5 × 5. Here, we use the KV kernel, which is one of the commonly used kernels for preprocessing in traditional feature extractors, as shown below.

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

The hard wired layer aims to strengthen the weak stego signal, hence to provide a much better initialization to drive the whole network as compared with random initialization. The first and fifth convolutional layer have 16 filter kernels of size 5 × 5, and the second to fourth convolutional layers have 16 filter kernels of size 3 × 3. The filtering stride of all the convolution operation is set to 1. Meanwhile, overlapping average pooling operation is applied to each convolutional layer with window size 3 × 3 and stride 2. Finally, the extracted features from the convolutional layers are passed to two fully connected layers. Each of the two layers has 128 neurons, and the output of each neuron is activated by the Rectified Linear Units (ReLUs) [16].

The settings of architecture we use here is the same as in [19], except that the activation function used in the five convolutional layers in this work is a variant of Gaussian function that has a better performance than Gaussian function. It is shown as below.

$$f(x) = 1 - e^{-\frac{x^2}{\sigma^2}}, \tag{2}$$

where σ is a parameter that determines the width of the curve.

3.3 Constructing Auxiliary Tasks

In this section, we introduce how to construct auxiliary tasks that are used for incorporate prior knowledge into the training of steganalysis models. Firstly, these tasks should be related to the specific steganalysis task. Secondly, we expect that they would provide complementary information that are useful for steganalysis while are hard to capture by the target task.

In steganalysis, effectively discovering and exploiting dependencies among individual image elements is crucial for the detectors to obtain a good performance. In different methods, different types of image statistics are exploited to model the dependencies. In traditional steganalysis systems, features are extracted by computing global statistics such as high order co-occurrences from noise residuals. These global statistics have been proved to be efficient for steganalysis. Differently, the CNN model describe the relationships among a large number of image elements through multi stage filtering and pooling operations, and the features extracted here are more related to local statistics from a neighborhood. It means that the useful global statistics in conventional steganalysis methods are hard to be captured by the CNN model. Hence, it is desirable that the auxiliary tasks can encode the global statistic information to the CNN model. To this end, we propose constructing the auxiliary tasks by computing features with a traditional method as mentioned before. In our experiments, we use a 169 dimensional feature vector formed from the noise residual computed using the KV kernel as auxiliary outputs. The detailed feature extraction step can be found in [5].

4 Experiments

To evaluate the effectiveness of the proposed framework, we conduct experiments on the BOSSbase 1.01 dataset [2], which contains 10,000 images acquired by seven digital cameras in RAW format and subsequently processed to the size of 512×512.

In our experiments, to further improve the network's generalization ability and to reduce the effects of overfitting during training, two commonly used regularization techniques are used in the CNN architecture. Firstly, the technique called "dropout" as detailed in [7] is applied for regularizing the two fully connected layers. Secondly, we take advantage of the data augmentation skill to artificially enlarge the dataset to reduce overfitting problem. It is applied by extracting random 256×256 patches as well as their flip version from the 512×512 images, and training the network on these extracted patches. At testing time, five 256×256 patches, including the four corner patches and the center patch, and their flip version are extracted. The network makes a prediction on each of these patches, and averages the ten predictions to produce a more robust estimate of the class probabilities.

Table 1 Detection error of different methods on BOSSbase 1.01

bpp	WOW			S-UNIWARD		
	0.3 (%)	0.4 (%)	0.5 (%)	0.3 (%)	0.4 (%)	0.5 (%)
SRM + Ensemble	25.57	20.90	16.60	26.12	20.92	16.70
CNN	28.93	21.98	17.35	32.03	24.20	20.65
Proposed	24.18	19.30	16.0	29.58	22.33	17.38

The proposed models are implemented using the code provided by Krizhevsky et al. [12], which allowed for rapid experimentation. We use a Tesla K40c GPU with 12GB of memory and two Tesla K20m GPU with 5GB of memory. In the overall loss function, the weight for the target loss is set to 1, and the weight for the auxiliary loss is set to 0.005 empirically. All the trainable parameters in the network are initialized randomly and trained by back-propagation algorithm as has been mentioned.

Table 1 shows the comparison of our results with two other methods. The detection error $P_E = min_{P_{FA}} \frac{1}{2}(P_{FA} + P_{MD}(P_{FA}))$ is used to evaluate the performance the these methods, where P_{MD} is the missed detection rate and P_{FA} is the false alarm rate. The "CNN" means the method proposed in [19]. In that work, the CNN model is trained without model regularization from auxiliary tasks. Here, for fair comparison, the CNN architectures in our proposed method and this method use the same settings. The "dropout" and the data augmentation are applied to both methods. The "SRM + Ensemble" means the method that based on training a ensemble classifier on SRM feature set, which is one of the representative traditional steganalysis schemes. The experiments are run on two content-adaptive steganographic algorithms, WOW [8] and S-UNIWARD [9], with three payloads respectively. From Table 1, we can observe that the proposed method achieves 1–4 % improvement in detection error over the "CNN" method in [19]. It means that model regularization via transfer learning from auxiliary tasks is helpful for learning features in steganalysis. And the detection performance for the WOW algorithm is better than the "SRM + Ensemble" method, which is one of the state-of-the-art methods in image steganalysis.

5 Conclusion

In this paper, we propose a new CNN based framework to effectively learn feature representations for steganalysis. In the framework, we use transfer learning from auxiliary tasks to encode priori knowledge into the learning process of CNN models. This would provide a good model regularization for improving the training of CNN. we construct auxiliary tasks by computing features to capture global image statistics which are useful for steganalysis but hard to be seized by the CNN network structure. Experimental results show the effectiveness of the proposed frame work on improving feature learning using CNN models for steganalysis. We also achieve a better performance on detecting the WOW algorithm against the traditional steganalysis scheme that using SRM feature set.

Acknowledgments This work is funded by the National Nature Science Foundation of China (Grant No.61303262).

References

1. Ahmed A, Yu K, Xu W, Gong Y, Xing E (2008) Training hierarchical feed-forward visual recognition models using transfer learning from pseudo-tasks. In: ECCV, pp 69–82
2. Bas P, Filler T, Pevný T (2011) Break our steganographic system: the ins and outs of organizing boss. In: Information hiding, pp 59–70
3. Ciresan D, Meier U, Schmidhuber J (2012) Multi-column deep neural networks for image classification. In: IEEE conference on computer vision and pattern recognition, pp 3642–3649
4. Denemark T, Sedighi V, Holub V, Cogranne R, Fridrich J (2015) Selection-channel-aware rich model for steganalysis of digital images. In: National conference on parallel computing technologies (PARCOMPTECH), pp 48–53
5. Fridrich J, Kodovsky J (2012) Rich models for steganalysis of digital images. IEEE Trans Inf Forensics Secur 7(3):868–882
6. Gul G, Kurugollu F (2011) A new methodology in steganalysis: breaking highly undetectable steganograpy (hugo). In: Information hiding, pp 71–84
7. Hinton GE, Srivastava N, Krizhevsky A, Sutskever I, Salakhutdinov RR (2012) Improving neural networks by preventing co-adaptation of feature detectors. arXiv:1207.0580
8. Holub V, Fridrich J (2012) Designing steganographic distortion using directional filters. In: The IEEE international workshop on information forensics and security, pp 234–239
9. Holub V, Fridrich J (2013) Digital image steganography using universal distortion. In: Proceedings of the first ACM workshop on Information hiding and multimedia security, pp 59–68
10. Holub V, Fridrich J (2013) Random projections of residuals for digital image steganalysis. IEEE Trans Inf Forensics Secur 8(12):1996–2006
11. Ji S, Xu W, Yang M, Yu K (2013) 3D convolutional neural networks for human action recognition. IEEE Trans Pattern Anal Mach Intell 35(1):221–231
12. Krizhevsky A (2012) Cuda-convnet. http://code.google.com/p/cuda-convnet/
13. Larochelle H, Bengio Y, Louradour J, Lamblin P (2009) Exploring strategies for training deep neural networks. J Mach Learn Res 10:1–40
14. LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proc IEEE 86(11):2278–2324
15. Mobahi H, Collobert R, Weston J (2009) Deep learning from temporal coherence in video. In: Proceedings of the 26th annual international conference on machine learning, pp 737–744
16. Nair V, Hinton GE (2010) Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th international conference on machine learning, pp 807–814
17. Oquab M, Bottou L, Laptev I, Sivic J (2014) Learning and transferring mid-level image representations using convolutional neural networks. In: IEEE conference on computer vision and pattern recognition, pp 1717–1724
18. Pevny T, Bas P, Fridrich J (2010) Steganalysis by subtractive pixel adjacency matrix. IEEE Trans Inf Forensics Secur 5(2):215–224
19. Qian Y, Dong J, Wang W, Tan T (2015) Deep learning for steganalysis via convolutional neural networks. In: IS&T/SPIE electronic imaging, pp 94,090J–94,090J
20. Salakhutdinov R, Hinton GE (2009) Deep boltzmann machines. In: International conference on artificial intelligence and statistics, pp 448–455
21. Shi YQ, Chen C, Chen W (2007) A markov process based approach to effective attacking jpeg steganography. In: Information hiding, pp 249–264
22. Shi YQ, Sutthiwan P, Chen L (2013) Textural features for steganalysis. In: Information hiding, pp 63–77

23. Tang W, Li H, Luo W, Huang J (2014) Adaptive steganalysis against wow embedding algorithm. In: Proceedings of the 2nd ACM workshop on information hiding and multimedia security, pp 91–96
24. Wan L, Zeiler M, Zhang S, Cun YL, Fergus R (2013) Regularization of neural networks using dropconnect. In: Proceedings of the 30th international conference on machine learning, pp 1058–1066
25. Zeiler MD, Fergus R (2013) Stochastic pooling for regularization of deep convolutional neural networks. arXiv:1301.3557