AG-Net: An Advanced General CNN Model for Steganalysis

HAN ZHANG [1,2], FUXIAN LIU[2], ZHIHUA SONG [1], XIAOFENG ZHANG[1], AND YONGMEI ZHAO [1,3]

[1]College of Equipment Management and UAV Engineering, Air Force Engineering University, Xi'an 710038, China
[2]Air and Missile Defense College, Air Force Engineering University, Xi'an 710051, China
[3]School of Computer Science and Engineering, Northwestern Polytechnical University, Xi'an 710072, China

Corresponding author: Zhihua Song (ola_alo@126.com)

ABSTRACT Steganography has made great progress over the past few years due to the advancement of deep convolutional neural networks (DCNNs), which have been successfully used to multi-domains. Correspondingly the performance of steganalysis models inevitably encounters a bottleneck since the CNN based steganography models perform better. In this paper, we propose an Advanced General convolutional neural Network (AG-Net) for steganalysis to deal with this problem. We firstly design a confrontation module to extract and compare features of cover and stego images, which are captured from an unknown steganography network. Then, we construct the association between two adjacent confrontation modules according to the feature comparison of the previous module, to accumulate the differences of mid-and high-level features between the cover and stego images. Thirdly, we deliver the loss of the last confrontation module to a softmax layer after batch normalization and scalarization, to classify and detect stego images. Extensive experiments and evaluations demonstrate that the proposed AG-Net can achieve promising performance in response to different challenging steganographic algorithms.

INDEX TERMS General steganalysis, convolutional neural network, confrontation module.

I. INTRODUCTION

Steganalysis is a method to judge whether the object of analysis contains secret information when the stego is known to use the carrier but the stego method is unknown. It is used to distinguish the objects of cover and stego, namely, the binary classification problem. At present, the research focuses on digital images as the carrier of hidden information, mainly with respect to the steganalysis method and technology of digital images [1].

Steganalysis belongs to the category of pattern recognition. Due to the successful application of deep learning in the field of pattern recognition, steganalysis technology has been developed. Steganalysis is divided into two stages: feature construction and classifier. Since the steganalysis operation modifies the high frequency signal of the image, in the feature construction stage, the high pass filter is often used to calculate the residual image, and then various statistical models are used to extract the features of steganalysis. Good features play a key role in the detection accuracy of steganalysis [2].

However, the analytical characteristics of traditional pairs are mainly based on artificial experience and heuristic design, which requires knowledge in specific fields [3]. The feature construction and classifier training of steganalysis are carried out independently, so it is difficult to achieve synchronous optimization between feature design and classifier. Aiming at the above problems of traditional steganalysis [4], this paper combines deep learning with steganalysis and uses the deep learning model to obtain the simulation complex representation of the model to achieve the purpose of automatic learning for effective feature expression. At the same time, using the end-to-end learning process of deep learning, the feature construction and classifier are completed synchronously in a structure. If the optimization cannot be synchronized, it means that the classification guidance cannot be used to capture useful information in the feature extraction step.

Among many deep learning models, convolutional neural networks are the most representative and the most commonly used network model in steganalysis [5]. At present, the steganalysis method based on convolutional neural networks mainly uses two convolution layers, three convolution layers or five convolution layers to design the network.

The associate editor coordinating the review of this manuscript and approving it for publication was Shenghong Li.

In 2014, Tan *et al.* first proposed a steganalysis method based on deep learning, and demonstrated the potential of CNN in the steganalysis field for the first time [10]. The Xu-Net network improves the feature learning ability [20]. The Ye-net network introduces channel selection in deep learning for the first time [18]. Chen, Xu *et al.* introduced steganalysis of JPEG images in 2017 [17]. Zeng *et al.* proposed a hybrid deep learning steganalysis model for steganalysis of JPEG images [3]. These are steganography methods for spatial domain and frequency domain. In 2014, Goodfellow *et al.* proposed a kind of Generative Adversarial Network (GAN) [24], that aims to promote the evolution of all parties by using game confrontation between two sides in the information hiding field, making the GAN popular [7].

With the continuous development of deep learning, convolutional neural network has achieved good results in steganalysis, but these results are achieved under the condition of a high embedding rate, and the detection effect is not good. Although the detection effects of some results under the condition of a low embedding rate are improved [8], this is not suitable for steganalysis of JPEG images. Follow-up research will change the network structure, number of layers, activation function and addition of filters are studied in order to improve the non-universality of steganalysis and improve the detection rate.

Therefore, in the framework of deep learning, we propose an improved general convolution neural network model, called AG-Net, for steganalysis. The main contributions and works are summarized as follows:

(1) We firstly design a confrontation module to extract and compare the features extracted from the cover part and the stego part, which were originally intercepted from any kind of steganography network.

(2) We then construct the association between two adjacent confrontation modules according to the feature comparison of the previous module, to accumulate the differences of mid- and high-level features between the cover and stego parts of the previous module.

(3) We thirdly deliver the loss of the last confrontation module to a softmax layer after batch normalization and scalarization, to classify and detect stego images.

A large number of experiments show that the proposed algorithm achieves better performance in challenging steganography, and it outperforms existing techniques in terms of error probability in different steganography. It can improve the generality and accuracy of steganalysis to a certain extent. The rest of the paper is arranged as follows. After a brief review of CNN, in Section II, we describe the proposed AG-Net steganalysis model. In Section III, the experimental results are presented. Finally, Section V discusses the conclusion and future work.

II. RELATED WORKS

The structure of the convolutional neural network model is very similar to the traditional steganalysis process. The first step of steganalysis is residual acquisition, which is composed of several high-pass filters to obtain the residual

TABLE 1. Improved Xu-Net.

Xu-net	Improvement measures	Implementation steps
	add absolute value layer	after the first convolution
	Tanh Activation function	used in the first two layers of network
	1 * 1 convolution kernel	in layer 3-5 network

image (the noise signal of the image). The first layer of the convolution neural network is a convolution layer, which is similar to a low-pass filter to obtain a feature map (the key content of the response image). The second and third steps of steganalysis are truncation and quantization, respectively, where the purpose is to reduce the residual range and retain the main residual information. The convolution layer of the convolution neural network is the activation function and pooling layer, whose functions are nonlinear transformation and dimensionality reduction of the feature map, respectively. After residual processing, steganalysis must summarize and reduce the residuals by certain statistical methods to obtain the features, while the convolutional neural network, at the end of the feature representation, obtains the merged feature graph through several fully connected layers. At the end of the steganalysis and convolution network, a classifier is connected to output the classification results. In the second part, we will review the existing steganalysis methods based on deep learning and focus on their advantages and disadvantages.

A. SPATIAL DOMAIN STEGANALYSIS

Spatial steganography is a method of hiding secret information within an image by processing the different pixel values of the covered image, thus affecting the statistical characteristics of the image. Next, we focus on the main contribution of the deep learning method in spatial steganalysis and compare different machine learning methods [9]. In 2014, Tan and Li first proposed a network architecture for steganalysis based on deep learning. Tan and Li [10] proposed the first CNN structure for the steganalysis of digital images in the spatial domain. The similarities between SRM and CNN are comparable to spam. The network has a four-layer convolutional neural network structure. In addition, its average pool is better than its maximum pool. However, the FC layer is slow because it is too large. The network uses three models (three training modes) to detect the Hugo steganalysis algorithm; its embedding rate is 0.4 bpp, and the error rate is 48%. Spam and RAM reflect the potential of CNN in the steganalysis field. Although the proposed model performs better than SPAM [21], it remains inferior to SRM [11]. Moreover, it reduces the dimensionality of features and becomes a useful tool in steganalysis. The generated network is shown in Table 1:

In 2015, Qian [12] *et al.* first proposed a network model combining steganalysis with deep learning. Its advantages are

restoration B/W steganalysis and GCNN. This model is better than spam and introduces a Gaussian activation function and HPF layer. The hybrid set function is used to improve the feature learning ability of the network front end, and the activation function ReLU is used to reduce the difficulty of backpropagation. The global average pool and $1 * 1$ convolution kernel are used in the back end of the network to prevent overfitting and information loss. However, the performance of this network is inferior to that of SRM, and their networks will not converge without a high pass filter. BN is increased to improve the convergence rate and avoid falling into local minima. The performance of this CNN network for steganalysis is comparable to that of the spatial rich model [13]. This paper not only emphasizes the role of a high pass filter in network convergence but also proves that transfer learning is conducive to detecting a low embedding rate steganography framework. In 2016, Ren *et al.* [14] designed a CNN-based steganalysis network with poor performance but higher detection accuracy.

In the same year, Holub and Fridrich [15] added many new network structures on the basis of Qian-Net. This network uses an absolute ABS activation layer, BN (batch normalization) and pool layer. Its primary advantage is that it not only ensures more effective learning characteristics but also effectively avoids overfitting and other problems and improves the accuracy of SRM [11]. The absolute ABS activation layer is used to generate the feature map generated by the first convolution layer. It can learn more efficient features that may help avoid overfitting problems.

B. JPEG IMAGE STEGANALYSIS

Zeng *et al.* [16] proposed a hybrid deep learning steganalysis model for detecting steganalysis methods of JPEG images, but its network performance cannot be compared with PHARM. Xu *et al.* [17] proposed a 20-layer deep learning neural network for JPEG steganalysis: that is, a fast connection technique. This network framework uses a convolution layer instead of a pooling layer, which improves the detection accuracy to a certain extent. Compared with Zeng *et al.*, the error rate is reduced by 35%. Chen then proposed an improved deep learning framework based on the JPEG compression algorithm based on Xu-Net and achieved good network performance. Wu *et al.* [18] proposed a deep residual network framework for detecting adaptive JPEG steganography. The layer number of the network architecture is deeper than before. Although batch normalization is not effectively used in the network, the layer number of the network exerts a very obvious effect on capturing the statistical characteristics of digital images. Huang proposed a ResDet with a high embedding rate. In 2018, Yedroudj-Net [18] proposed a CNN that outperforms the state-of-the-art methods in terms of error probability. In 2020, Eslam M discards the pre-processing step, to reduces the computational cost [28].

Other networks were proposed in 2017 for JPEG image steganalysis. In [16], the authors proposed a hybrid deep learning steganalysis model for detecting steganalysis methods in JPEG images. At present, JPEG based on deep learning is mainly realized by modifying the coefficient of variation. After inserting the secret information, the digital image is saved and transmitted through the network. The use of this file format is the most extensive and prominent. When compressed to one-eighth of the original image size, the image of this format has high visibility. However, compared with other weak signals, hidden information is difficult to find in JPEG compressed images.

Steganalysis researchers have applied deep learning technology to networks and achieved many successful results. We can further improve the accuracy of the deep learning framework by analyzing and comparing different network designs, fusion and learning strategies, and deep learning skills and introducing prior domain knowledge into the CNN architecture.

Convolutional neural networks have been continuously functioning in many directions and have achieved breakthroughs in speech recognition, face recognition, general object recognition, motion analysis, natural language processing and even analysis. For steganalysis, many studies have shown that convolutional neural networks have achieved very successful results. However, there are still two problems to be solved: the practicability of steganalysis and the accuracy of steganalysis are discussed.

III. PROPOSED AG-NET STEGANALYSIS FRAMEWORK

In this section, we propose a novel general CNN structure named ''AG-Net'' to improve the accuracy of spatial domain steganalysis. The core idea of the proposed AG-Net is to extract the features of cover and stego layer by layer and compare the differences. Each time, the comparison model level is enlarged for classification and identification. We design an improved frontal confrontation module to extract and compare the cover and stego images in the location steganography network and then guide the next group of modules through the output of the first module to construct the association between the two modules. Finally, after batch standardization and scalarization, steganography is used to classify and recognize steganalysis images. The core idea of the proposed AG-Net is to extract the features of cover and stego layer by layer and compare the differences. Each time, the comparison model level is enlarged for classification and identification. The architecture of the proposed AG-Net is shown in Figure 1.

A. PRE-PROCESSING MODULE

The preprocessing module is designed to process the input cover and stego images with convolutional blocks to filter residual noise components of images. The convolutional block is composed of a predefined high-pass convolutional layer with kernel, stride 1 and padding 2. By initial convolution, the preprocessing module can narrow dynamic ranges to suppress input images and promote the signal-noise ratio, which is computed by the weak stego signal and the strong image signal. We designed the preprocessing module according to previous studies [19], which certified that networks without preprocessing blocks converge more slowly and
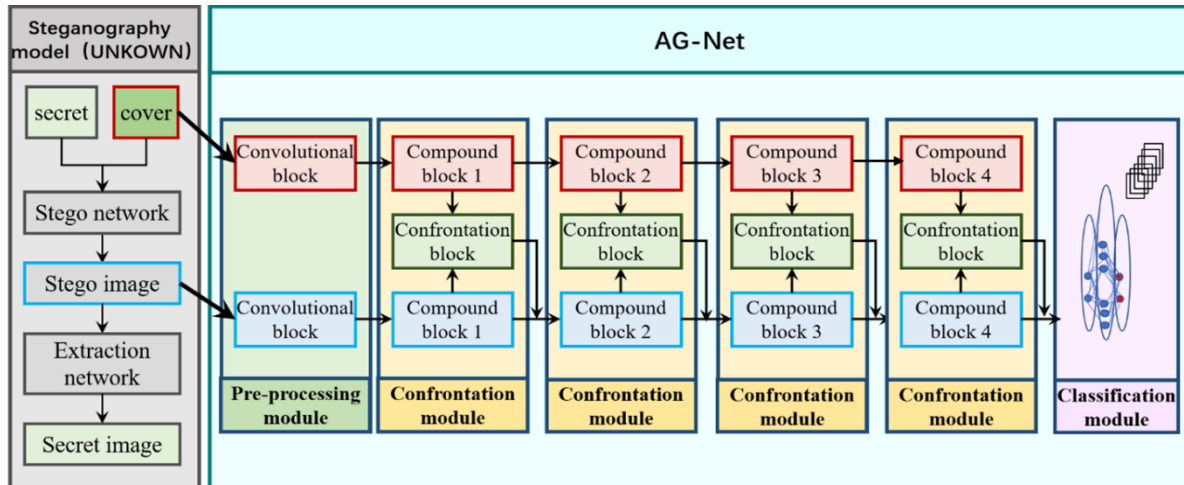
FIGURE 1. Architecture of an Advanced General CNN model for Steganalysis. On the left side of the image is a general steganographic network, the process is unknown, and it is applicable to any framework. All we know is cover image and stego image. Therefore, the input objects of steganalysis network designed by us are only these two kinds of images, and there is no need to pay attention to other parameters from steganography network. On the right side is the stego analysis framework we designed. The input cover image and stego image are classified and identified by using softmax through the hierarchical progression of multiple comparison modules.
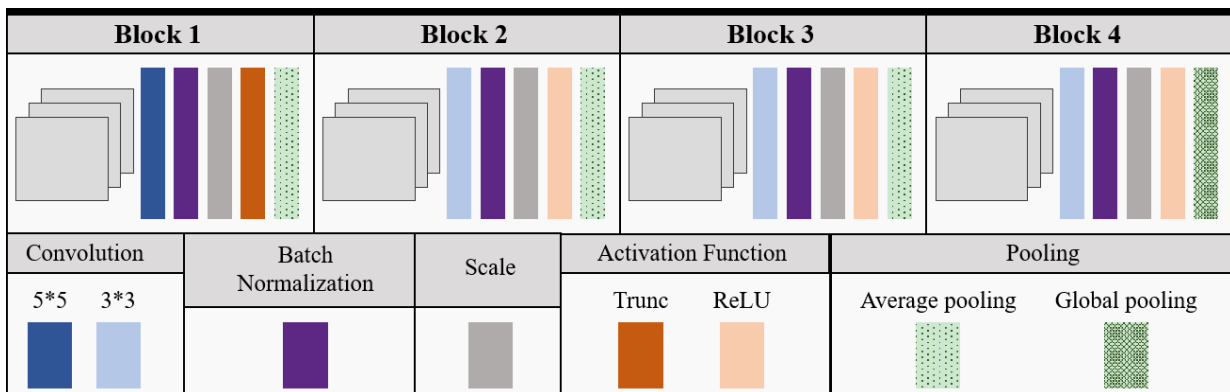
FIGURE 2. Architecture of compound blocks in different confrontation modules. All four blocks, both for the cover and stego branches, are composed of a convolutional layer, batch normalization layer, scale layer, activation layer and pooling layer. The modes of the activation and pooling layers for the four compound blocks differ according to their positions and functions.

networks with preprocessing modules can express signals in a more compact and robust way.

As shown in Figure 1, the inputs of pre-processed module are the cover and stego images, for the cover and stego processing branches respectively, and the outputs of the module are convolutional images, feeding the remainder network. Since images are resized to 256 × 256, the output is 30 × 256 × 256 for convolutional block of each branch.

B. CONFRONTATION MODULE

The confrontation module is composed of two compound blocks (for the cover and stego branches) and a confrontation block, as shown in Figure 1. Designed for extracting deeper features, the compound block contains a convolutional layer, followed by a batch normalization layer, a nonlinear activation layer, and a pooling layer. The confrontation is designed to compare the features of cover and stego branches.

The difference between the two branches is delivered to the next module to inspire learning of more contrastive features. As shown in Figure 1, the inputs of the confrontation module are the features learned by the cover and stego block of the previous module and the difference extracted by the confrontation block of the previous module (for the first confrontation module, no difference is delivered).

Considering the size of input images, we design four sequential connected confrontation modules, with different sizes for compound blocks of the four modules. The architectures of different compound blocks are shown in Figure 2. The parameters of different compound blocks are shown in Table 2.

As shown in Figure 2, all four blocks are composed of a convolutional layer, batch normalization layer, scale layer, activation layer and pooling layer. The function and parameter of every layer of the compound block are analyzed as follows:

TABLE 2. The parameters of different compound blocks.

Block	input	convolution	stride	pad	output
1	256×256	$30@(5 \times 5 \times 30)$	1	2	128×128
2	128×128	$30@(3 \times 3 \times 32)$	1	1	64×64
3	64×64	$32@(3 \times 3 \times 64)$	1	1	32×32
4	32×32	$64@(3 \times 3 \times 128)$	1	1	1×1

1. A convolution layer. We set the size of convolutional layers according to Xu-Net [20], with kernel size 5×5 for Block 1 and 3×3 for Blocks 2, 3 and 4. The convolutional layers have no bias terms since they are moved to the scale layer.

2. A batch normalization (BN). The BN layer is used to normalize the distribution of each feature as described in [21]. The normalized features have a zero mean and a unit variance. By using a BN layer, the block can desensitize the training parameters for initialization [22], allowing us to speed up the learning with a larger learning rate and improving the classification accuracy.

3. A scale layer. The scale layer is a bias term separated from the convolutional layer as an independent part. Similar to ResNet [23], the scale layer attempts to learn more efficient translation parameters. Separating the scale function with the convolutional layer can enable learning of more typical features.

4. A nonlinear activation layer. The nonlinear activation layer is used to transform the inseparable features to a separable space. We employ the truncation function (Trunc) for Block 1 to narrow the range of data values, which are sparse and not statistically significant, to prevent deeper layers from modeling large values. The formula of Trunc is given in Eq. 1.

$$Trunc\,(x) = \begin{cases} T, & x > T \\ x, & -T \leq x \leq T \\ -T, & x < -T \end{cases} \quad (1)$$

The nonlinear activation function of the other three blocks is ReLU. The ReLU can avoid vanishing and exploding gradients since it is sparse. The formula of ReLU is given in Eq. 2.

$$ReLU\,(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (2)$$

5. An average pooling layer. We use average pooling in Block 1, 2 and 3, and global average pooling in Block 4. The benefit of average pooling is down-sampling features, thus to reduce dimensionality. Compared with general average pooling, global average pooling can transport feature maps of any size to one by one element, to avoid grasping the embedded pixel information [24].

As shown in Figure 2 and Table 2, the parameters of different blocks differ according to their positions and functions. The input size of the first compound block of cover and stego branches is the output of the corresponding preprocessing

module. In the process of graphics processing, a fixed high pass filter is used for image preprocessing to obtain the corresponding residual image. When using different convolution cores to process images, we must ensure that the output residual images obtained by filtering are acceptable. Among the four convolution layers, the number of kernels in each layer is set to 30, 30, 32 and 64, and the kernel size is set for Block 1 and for the other three blocks. The stride size is 1, and the padding size is 2 for Block 1 and 1 for the other three blocks. The stride and padding size are set to ensure that the size of the output is half that of the input.

The confrontation block is designed to learn the difference between the cover and stego branches. Therefore, the confrontation block is a D-value computation for the convolutional kernels in the cover and stego blocks of the same confrontation module. Obviously, the input of the confrontation block is the convolutional kernels of the cover and stego blocks, and the output is the D-value between the kernels of the two branches.

The confrontation module is the core of the entire network. The compound block can learn deeper features by its convolutional layer and achieve efficiency and robustness by the batch normalization layer, scale layer and pooling layer. The confrontation block can learn the difference between the cover branch and stego branch. The difference can also be accumulated by delivery to the next confrontation module, such that the learned features would be more representative.

C. CONNECTION BETWEEN CONFRONTATION MODULES

The output of a previous confrontation module contains the features learned by the two compound blocks and the difference learned by the confrontation block. The features of compound blocks are delivered to the corresponding compound block of the following confrontation module. The difference of the previous module is delivered to the compound block of stego branch only, to initialized the feature maps of the convolutional layer. The difference delivered to the following stego compound block is used to guide the previous module lost to the following confrontation module, and further emphasize the difference between the middle and higher-level features of the cover branch and stego branch.

It is worth mentioning that the difference of the previous module has the same size with the convolutional kernel of the compound block. When the difference is used to initialize the convolutional kernel of the following stego compound

branch, there may has a problem that the sizes are not match. For example, the size of the difference of Block 1 is 5×5, and the size of the stego compound branch of Block 2 is 3×3. When it comes to this problem, we select the center part of the previous one to match the convolutional kernel of the following stego compound branch. So as to construct the association between two adjacent confrontation modules.

D. CLASSIFICATION MODULE

The main idea of this module is to classify the features of the fourth confrontation module. The classification module has three layers of full connection. The first two layers of full connection contain 128 nodes, and the activation function is ReLU. The last layer of full connection contains two nodes, and the activation function is softmax. Since many papers have described this function in detail, the readers are referred to the original papers [29] for more information.

### IV. SIMULATION AND TRAINING ENVIRONMENT

We conduct several experiments to evaluate the effectiveness of the proposed AG-Net in this section. First, we introduce the dataset and different steganography methods used in experiments. Then, we present the parameters used in the training and test process. Third, the experimental results are displayed and analyzed in detail.

A. DATASET AND STEGANOGRAPHY METHODS

In this research, experiments are conducted on a commonly used publicly available source BOSSBase. BOSSBase is a standard database in steganalysis with 10000 images. All of the images are selected from seven types of cameras. The cover image is gray and formed by zooming, clipping and other operations in the format of PGM. The size of images is 512×512. We selected some categories of BOSSBase and show them in Figure 3. Figure 3 contains different scenes such as architecture, and scenery. In terms of texture characteristics, when the image is magnified 200 times, more changes of the image will be focused on the position with complex texture.

We use three well-known content adaptive steganography algorithms, HILL [25], WOW and S-UNIWARD, for the embedding of secret images in the spatial domain. All three steganography algorithms are carried out with an STC simulator using publicly available codes. It is worth mentioning that we use MATLAB as our platform instead of C++ to create a random embedding key, which can prevent overfitting in the training set and gain a better generalization capability. In contrast, deep learning-based steganography algorithms could achieve extraordinary classification performance with a fixed embedding key created on the C++ platform. This similar phenomenon was proven in [26], where the authors created stego images with different embedding keys.

Our network was trained on a PC with an Intel Xeon (R) bronze 3104 CPU, operating speed of 1.7 GHz, Titan x GPU card, 32 GB DDR4 memory, NVIDIA GeForce RTX2080Ti graphics processing unit (GPU), and memory of 11 GB. To further verify the feasibility of this method,
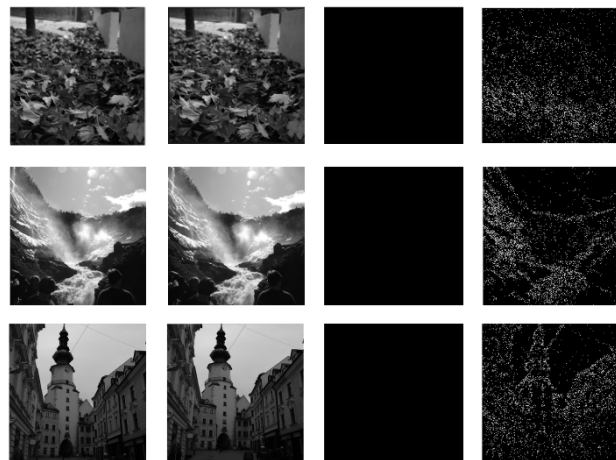
FIGURE 3. Samples of several categories in BOSSBase dataset. Images have different pattern when magnified to different times. The first column is the original image. The second column is a dense (noisy) image. The third column is the difference value, which is the interpolation image caused by the difference value that has not been magnified, which is completely black. The fourth column is the image magnified 30 times.

our configuration is based on the common PyTorch toolkit experimental environment [27]. The open-source operating system of this paper is based on the deep learning framework of the software.

B. PARAMETERS IN TRAINING AND TEST PROCESS

Since the input size of the proposed AG-Net is 256×256, we resampled all the 512×512 images of BOSSBase to 256×256. For training and testing, we split the resized BOSSBase into two sets, 50% for cover and 50% for stego. Both for cover and stego sets, 4000 out of the 5000 images are randomly selected for training, and the remaining are set for validation. We mainly evaluate the performance of all methods under three kinds of payloads: 0.3, 0.4 and 0.5 bit/pixel (BPP). Note that AG-Net is trained with different embedding schemes at different payloads, for tasks of varying difficulties, which accumulates and guide the decay of the learning rate.

C. EXPERIMENTAL RESULTS AND ANALYSIS

We conducted several experiments with state-of-the-art deep learning based steganalysis algorithms for performance comparison and show the simulation results in Table 3.

As shown in Table 3, all the models performs better when against with S-UNIWARD, than with WOW and HILL. That is because the secret messages are embedded at more complex regions in a cover image by WOW and HILL than that by S-UNIWARD steganographic. Obviously, the detection error rates of the Xu-Net, Ye-network, SN-network, ReST-Net and the proposed AG-Net on three steganographic algorithms at three payloads. Compared with the Xu-Net, Ye-network, SN-network and ReST-Net steganalysis, proposed model obtains significant performance against with different steganographic algorithms. This result is possibly due to the following reasons: (1) the confrontation module

TABLE 3. Performance comparisons between propsed AG-Net and other deep learning based steganalysis algorithms.

Steganography	Model	0.3bpp	0.4bpp	0.5bpp
	Xu-Net[20]	27.80%	20.70%	17.26%
	Ye-network[18]	26.54%	23.20%	17.61%
WOW	SN-network[30]	25.04%	21.03%	17.15%
	ReST-Net[31]	25.53%	20.60%	16.83%
	AG-Net	22.43%	16.38%	14.18%
	Xu-Net[20]	26.32%	19.88%	16.46%
	Ye-network[18]	25.62%	22.64%	17.64%
S-UNIWARD	SN-network[30]	20.71%	16.53%	12.71%
	ReST-Net[31]	23.56%	15.72%	13.83%
	AG-Net	19.34%	14.51%	10.73%
	Xu-Net[20]	26.86%	21.31%	18.18%
	Ye-network[18]	27.98%	23.08%	21.14%
HILL	SN-network[30]	23.60%	19.87%	16.29%
	ReST-Net[31]	24.84%	19.75%	16.53%
	AG-Net	22.04%	18.05%	15.40%

can capture the most represented features; (2) the difference accumulates through adjacent confrontation modules.

V. CONCLUSION

In this paper, we proposed an advanced general convolutional neural network for steganalysis, which is shown to perform better for general steganography methods. The experiments showed that AG-Net performs better than other state-of-the-art algorithms. Compared with other algorithms, AG-Net includes a confrontation module to compare and accumulate the differences between the cover and stego branches. Interesting avenues for future work include balancing efficiency and effectiveness.

REFERENCES

[1] H. Sajedi and M. Jamzad, "A steganalysis method based on contourlet transform coefficients," in *Proc. Int. Conf. Intell. Inf. Hiding Multimedia Signal Process.*, Aug. 2008, pp. 245–248.
[2] L. Pibre, J. Pasquet, D. Ienco, and M. Chaumont, "Deep learning is a good steganalysis tool when embedding key is reused for different images, even if there is a cover sourcemismatch," *Electron. Imag.*, vol. 2016, no. 8, pp. 1–11, 2016.
[3] H.-J. Zeng, Q. He, G. Liu, Z. Chen, B. Zhang, and W.-Y. Ma, "Query-based snippet clustering for search result grouping," U.S. Patent US20 060 026 152 A1, Nov. 10, 2006.
[4] J. Jalan, "Feature selection, statistical modeling and its applications to universal JPEG steganalyzer," Ph.D. dissertations, Iowa State Univ., Ames, IA, USA, 2009.
[5] H. H. Xiu, "Research on activation function in deep convolutional neural network," in *Proc. Conf. Artif. Intell. Healthcare*, 2020, pp. 19–24.
[6] Y. Qian, J. Dong, W. Wang, and T. Tan, "Learning and transferring representations for image steganalysis using convolutional neural network," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 2752–2756.

[7] K. Wang, C. Gou, Y. Duan, Y. Lin, X. Zheng, and F.-Y. Wang, "Generative adversarial networks: Introduction and outlook," *IEEE/CAA J. Automatica Sinica*, vol. 4, no. 4, pp. 588–598, 2017.
[8] J. Padhye, "Stepping-stone network attack kit (sneak) for evading timing-based detection methods under the cloak of constant rate multimedia streams," Ph.D. dissertations, Univ. Texas Arlington, Arlington, TX, USA, 2008.
[9] I. Hussain, J. Zeng, X. Qin, and S. Tan, "A survey on deep convolutional neural networks for image steganography and steganalysis," *KSII Transactions on Internet and Inf. Syst.*, vol. 14, no. 3, pp. 1228–1248, 2020.
[10] S. Tan and B. Li, "Stacked convolutional auto-encoders for steganalysis of digital images," in *Proc. Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA)*, Dec. 2014, pp. 1–4.
[11] J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 868–882, Jun. 2012.
[12] Y. Qian, J. Dong, W. Wang, and T. Tan, "Deep learning for steganalysis via convolutional neural networks," *Proc. SPIE*, vol. 9409, Mar. 2015, Art. no. 94090J.
[13] Y. Lizhe, X. Bo, and W. Xianjie, "BP network model optimized by adaptive genetic algorithms and the application on quality evaluation for class teaching," in *Proc. 2nd Int. Conf. Future Comput. Commun.*, May 2010, p. 273.
[14] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
[15] V. Holub and J. Fridrich, "Low-complexity features for JPEG steganalysis using undecimated DCT," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 2, pp. 219–228, Feb. 2015.
[16] J. Zeng, S. Tan, B. Li, and J. Huang, "Pre-training via fitting deep neural network to rich-model features extraction procedure and its effect on deep learning for steganalysis," *Electron. Imag.*, vol. 2017, no. 7, pp. 44–49, Jan. 2017.
[17] G. Xu, "Deep convolutional neural network to detect J-UNIWARD," in *Proc. 5th ACM Workshop Inf. Hiding Multimedia Secur.*, Jun. 2017, pp. 67–73.
[18] S. Wu, S. Zhong, and Y. Liu, "Deep residual learning for image steganalysis," *Multimedia Tools Appl.*, vol. 77, no. 9, pp. 10437–10453, May 2018.
[19] Y. Qian, J. Dong, W. Wang, and T. Tan, "Deep learning for steganalysis via convolutional neural networks," *Proc. SPIE*, vol. 9409, Mar. 2015, Art. no. 94090J.
[20] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22nd ACM Int. Conf. Multimedia*, 2014, pp. 675–678.
[21] L. Min, L. X. Ming, Y. Xue, Y. Yu, and W. Mian, "Improvement of universal steganalysis based on spam and feature optimization," in *Cloud Computing and Security*. Springer, 2016.
[22] C.-Y. Wu, N. Singhal, and P. Krähenbühl, "Supplementary material: Video compression through image interpolation," Tech. Rep.
[23] J. Fridrich and D. Soukal, "Matrix embedding for large payloads," *IEEE Trans. Inf. Forensics Security*, vol. 1, no. 3, pp. 390–395, Sep. 2006.
[24] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *Comput. Sci.*, pp. 1–11, Dec. 2014.
[25] J. Ni, J. Ye, and Y. I. Yang, "Deep learning hierarchical representations for image steganalysis," *IEEE Trans. Inf. Forensics Security*, vol. 12 no. 11, pp. 2545–2557, Nov. 2017.
[26] P. Bas, T. Filler, and T. Pevný, "Break our steganographic system: The Ins and outs of organizing BOSS," in *Information Hiding*, vol. 2011. Prague, Czech Republic: Springer-Verlag, May 2011.
[27] J. Yang, K. Liu, X. Kang, E. Wong, and Y. Shi, "Steganalysis based on awareness of selection-channel and deep learning," in *Proc. Int. Workshop Digit. Watermarking*. Cham, Switzerland: Springer, 2017, pp. 263–272.
[28] E. M. Mustafa, M. A. Elshafey, and M. M. Fouad, "Enhancing CNN-based image steganalysis on GPUs," *J. Inf. Hiding Multimedia Signal Process.*, vol. 11, no. 3, pp. 138–150, 2020.
[29] R. Zhang, F. Zhu, and J. Liu, "Efficient feature learning and multi-size image steganalysis based on CNN," 2018, *arXiv:1807.11428*.
[30] S. Wu, S.-H. Zhong, and Y. Liu, "Residual convolution network based image steganalysis with adaptive content suppression," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2017, pp. 241–246, doi: 10.1109/ICME.2017.8019304.
[31] S. Wu, S.-H. Zhong, Y. Liu, and M. Liu, "CIS-Net: A novel CNN model for spatial image steganalysis via cover image suppression," 2019, *arXiv:1912.06540*.

••••