mathematics

MDPI

Article

Image Steganalysis via Diverse Filters and Squeeze-and-Excitation Convolutional Neural Network

Feng Liu, Xuan Zhou, Xuehu Yan, Yuliang Lu * and Shudong Wang

College of Electronic Engineering, National University of Defense Technology, Hefei 230037, China; hemancute@163.com (F.L.); xzhou@secpol.net (X.Z.); publictiger@126.com (X.Y.); dongws@nudt.edu.cn (S.W.)
* Correspondence: publicLuYL@126.com

Abstract: Steganalysis is a method to detect whether the objects contain secret messages. With the popularity of deep learning, using convolutional neural networks (CNNs), steganalytic schemes have become the chief method of combating steganography in recent years. However, the diversity of filters has not been fully utilized in the current research. This paper constructs a new effective network with diverse filter modules (DFMs) and squeeze-and-excitation modules (SEMs), which can better capture the embedding artifacts. As the essential parts, combining three different scale convolution filters, DFMs can process information diversely, and the SEMs can enhance the effective channels out from DFMs. The experiments presented that our CNN is effective against content-adaptive steganographic schemes with different payloads, such as S-UNIWARD and WOW algorithms. Moreover, some state-of-the-art methods are compared with our approach to demonstrate the outstanding performance.

Keywords: steganalysis; convolutional neural network; diverse filter module; squeeze-and-excitation module

Citation: Liu, F.; Zhou, X.; Yan, X.; Lu, Y.; Wang, S. Image Steganalysis via Diverse Filters and Squeeze-and-Excitation Convolutional Neural Network. *Mathematics* **2021**, *9*, 189. https://doi.org/10.3390/math9020189

Received: 25 December 2020
Accepted: 18 January 2021
Published: 19 January 2021

1. Introduction

With the rapid development of information technology, covert communication methods using steganography have attracted increasing attention in recent years. With the improvement of steganography, it is more difficult to find out the embedding traces in objects, as the secret information is hidden in the texture area of the image with content-adaptive steganographic algorithms, such as HUGO [1], S-UNIWARD [2], WOW [3], HILL [4], MiPoD [5], JUNIWARD [6], UERD [7], ASO [8] and so on [9–11]. The core idea of these adaptive image steganography algorithms is to design the embedded distortion cost function, so as to separately measure the impact of each pixel modification in an image for the steganography security. It means that the security issues of steganography have been transformed into the issues of optimizing the distortion cost, which can guide the embedding operation of steganography by calculating the minimized embedding distortion to maximize the security of the steganography. As every coin has two sides, steganography could be easily exploited by criminals, so it is an important task to detect steganography.

The aim of steganography is to hide secret information in objects to covert communication. In contrast, steganalysis is to detect the hidden messages. However, steganalysis is a relatively challenging task as the changes of cover objects are almost impossible to be recognized by human eyes.

The traditional steganalytic schemes are usually based on well-designed hand-crafted features by expert experience matching with different machine-learning classifiers [12–16], for instance, the Subtractive Pixel Adjacency Matrix (SPAM) [13], the Spatial Rich Model (SRM) [14], ccPEV [17], DCTR [15] and their variants. In recent years, the popularity of Convolutional Neural Network (CNN) has promoted the study of steganalysis. The CNN schemes have shown great performance in image steganalysis. The spatial schemes for the current study include: Qian-Net [18], Xu-Net [19], Ye-Net [20], Yedroudj-Net [16],

ReST-Net [21], SRNet [22], Yedroudj-Net [23] and so on. All the above-mentioned CNN approaches can effectively distinguish steganographic images, and some even have better performances than the traditional approaches. After the analysis of the above-mentioned network structures, we can see that a typical steganalytic architecture mainly combines two elements of a feature extraction step and a classification step. One is to extract the noise residuals from the input image pairs as the features. The other is to classify the input image pairs into two classifications of covers and stegos. Although such approaches based on CNN have achieved a good performance in image steganalysis, the common thread in all these methods is that they use only one pipeline or do not combine the filters sufficiently.

In order to make full use of the diversity of filters, we propose an effective CNN architecture for steganalysis. There are two contributions of our method: the design of a diverse filter module (DFM) and squeeze-and-excitation module (SEM). Inspired by the Inception Network [24], which increases the width of the CNN structure, we use the DFMs to get more varied residual features. Similarly, we utilize the SEMs from learning the Squeeze-and-Excitation Network [25] to strengthen the key channels. Therefore, we named our CNN as "DFSE-Net" for steganalysis. The input image pairs can extract more features through multiple convolution kernels, at the same time the important feature maps can be highlighted. Therefore, the final fusion can obtain a better representation, and the experimental data set is BOSSBase1.01 [26]. Meanwhile, the experimental results demonstrate the outstanding performance of our convolutional network.

The rest of this paper is organized as follows. In Section 2, we review several related approaches of image steganalysis. Our CNN structure with DFMs and SEMs is presented in Section 3. In Section 4, the experimental results are reported. Section 5 concludes this paper.

2. Related Work

The first study of image steganalysis considering deep learning architecture was done by Tan and Li in 2014 with convolutional autoencoders [27], although the method was almost not effective, it was very innov ative at that time. In 2015, Qian et al. [18] proposed their GNCNN architecture and first imported a KV kernel as the image preprocessing layer. The design architecture could effectively improve the detection accuracy, but it was still not as good as traditional methods. A real breakthrough was achieved in 2016 by Xu et al. [19]. The performance of Xu-Net can be comparable with the traditional well-designed method consisting of SRM [14] and an ensemble classifier (EC) [28] for the first time. By analyzing its architecture of Xu-Net, we can know that they innovatively introduced the absolute activation (ABS) layer in the feature maps to facilitate the statistical modeling in the following layers; to prevent overfitting, they utilized the TanH function to limit the range of data values at early stages of their network, and used a 1×1 convolutional layer to construct a deeper network. In 2017, Ye et al. [20] presented their CNN architecture with a preprocessing layer containing thirty high-pass filters of SRM and designed an efficient activation function of truncated linear unit (TLU) that can better reveal the embedding artifacts. In 2018, Li et al. [21] firstly proposed a wide CNN architecture ReST-Net with diverse activation modules and parallel subnets. While the network only used one type of convolution kernel 3×3. In 2019, Zeng et al. [29] proposed a separate-then-reunion network for steganalysis of color images. These architectures have proven that the wider CNN can improve the detection performances. However, the architectures of parallel subnets with different sizes of combined kernels have not been extensively explored in steganalysis so far. This motivates us to design a wider CNN with different sizes of kernel units.

3. The Proposed Method

3.1. Architecture Overview

In order to make full use of the diversity of filters, DFSE-Net was designed and the overall structure is presented in Figure 1.

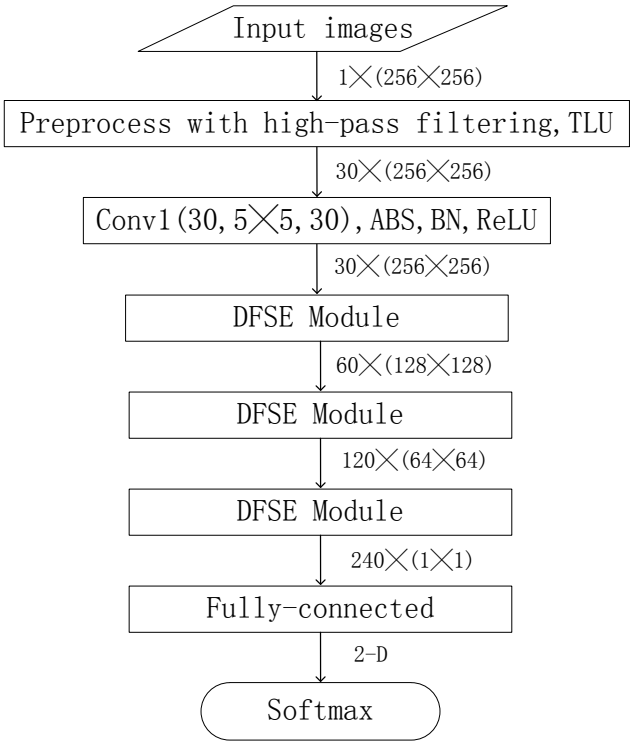

Figure 1. The architecture of DFSE-Net. For each convolutional layer, the data sizes are shown on the right side of each box and the types and parameters are displayed inside boxes.

Since DFMs can extract more diverse feature maps and SEMs can make our network focus on analyzing effective feature maps, the combination of two different modules can improve the detection effectiveness. The experimental results demonstrate our conclusion. In Figure 1, DFSE-Net consists of one image preprocessing layer, seven convolutional layers, with six convolutional layers in DFSE Modules, one fully-connected layer and one softmax layer. Due to the limitations of computing power, the size of the input image is 256×256.

The layer types and parameters are displayed inside boxes in Figure 1. $\mathrm{Conv}(x_1, a \times a, x_2)$ inside boxes means that the kernel size of the convolution layer is $a \times a$ and the number of input feature maps is x_1, the number of output feature maps is x_2. The full name of ABS is absolute activation, similarly, BN is batch normalization, TLU is truncated linear unit, ReLU is a rectified linear unit. The data sizes $(x \times (a \times a))$ denote the number and size of output feature maps, which are shown on the right side of each box.

The whole DFSE-Net can be simply divided into three steps. The first step is an image preprocessing layer with thirty high-pass filters of SRM [14], which can make our CNN concentrate on the embedding areas rather than the contents of images. Feature extraction is the second step, which consists of three DFMs and SEMs with seven convolutional layers. In this step, the feature maps are transformed into a 240-D feature vector. The third step is a linear classification module with one fully-connected layer and one softmax layer. In this step, the feature vectors are transformed into the output probabilities for each class. Each basic element is made of the following different layer types:

3.1.1. Convolution Layer

In our proposed architecture, we use three different convolution kernels instead of using only one type of 3×3 convolution kernel to extract local features of different sizes. In addition, the 3×3 and 5×5 kernels are parallelly computed in each DFM to capture more features. For the first convolution layer, the kernel size is 5×5, as to obtain a larger view of the local features. The 1×1 kernel is used after each SEM to integrate the rich

feature sets. The number of channels in each convolution layer is a comprehensive balance of computational complexity with network performance.

3.1.2. ABS Layer

The ABS layer [19] is only used after the first convolution layer. It discards the signs of the elements in the noise residuals, so that the statistical features of sign symmetry are forced to be considered in the feature maps. To show the performance of the ABS layer for image steganalysis, the comparisons are conducted based on the DFSE-Net with and without the ABS layer in the first convolution layer. Both models are trained for the WOW steganography algorithm at the payload of 0.2 bpp and 0.4 bpp. From Table 1, the DFSE-Net with ABS layer has a lower error rate of detecting the WOW steganography algorithm, and the ABS layer also accelerates the convergence and shows better performance, as shown in Figure 2.

Table 1. Steganalysis error rates comparisons of DFSE-Net with the absolute activation (ABS) layer and DFSE-Net without the ABS layer against the WOW steganography algorithm at 0.2 and 0.4 bpp. Both models are trained and tested on the BOSS dataset.

Algorithm	DFSE-Net with ABS Layer	DFSE-Net without ABS Layer
WOW (0.2 bpp)	0.247	0.275
WOW (0.4 bpp)	0.149	0.177

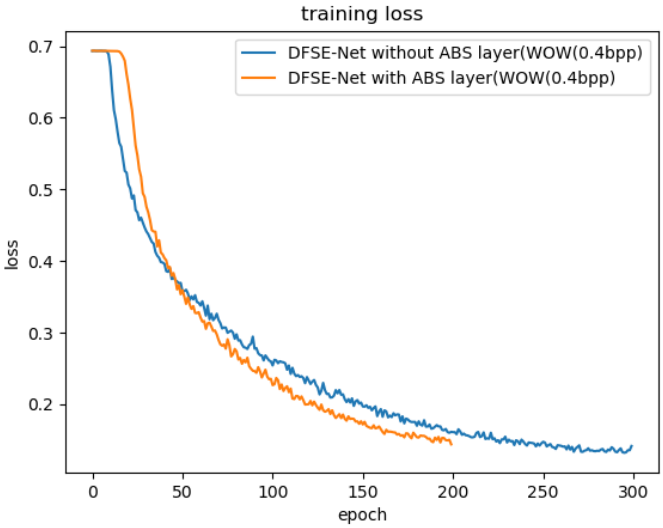

Figure 2. Comparing convergence performances of DFSE-Net with the ABS layer and DFSE-Net without the ABS layer against the WOW steganography algorithm at 0.4 bpp. Both models are trained and tested on the BOSS dataset.

3.1.3. BN Layer

The BN layer [30] is essentially a normalized network layer. It normalizes the distribution of each mini-batch to a zero-mean and a unit-variance. There are several advantages to using a BN layer. First, it can translate the distribution of the input feature maps. Second, it allows using a larger learning rate to speed up the learning, as it can desensitize networks to the initialization parameters. Third, it also can effectively prevent the gradient vanishing or exploding and overfitting in the training phase [30]. Hence, we choose to use the BN layer after each convolution layer in our proposed network.

3.1.4. Nonlinear Activation Layer

The activation layer introduces the nonlinearity into CNN networks, which can prevent gradient vanishing or exploding, increase the capability of feature representation

and so on. There are various types of activation functions that can be chosen, such as the conventional sigmoid function, ReLU (Rectified Linear Unit) function, hyperbolic tangent function, Truncated Linear Unit (TLU) function and so on. Among all of them, the ReLU function is commonly used in CNN and it can be formulated as Equation (1).

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases} \tag{1}$$

Except for the first layer in DFSE-Net, we apply the classical ReLU as the activation function in other blocks. Using the ReLU function after the conventional layer can make networks selectively respond to embedded signals among the input feature maps and conduct more efficient features. To a certain extent, the steganographic embedding procedure can be viewed as adding low-amplitude additive noises to cover images, and the embedding signals are usually in the range of [−1,1]. Therefore, we select the TLU, which is slightly modified from ReLU, in the first layer. As it contributes to the suppression of image contents and extraction of embedding signals more effectively. It can be defined as Equation (2).

$$Trunc(x) = \begin{cases} T, & x > T \\ x, & T \geq x \geq -T \\ -T, & x < -T \end{cases} \tag{2}$$

where $T > 0$ is the threshold determined by experiments. In this paper, the value of T is set to 3, the same as the value in Ye-Net [20].

To compare the performance of TLU with the ReLU function for image steganalysis, we conducted the comparisons based on the network shown in Figure 1. The DFSE-Net with TLU is trained in which the value of T is set to 3 in the first layer. The DFSE-Net with ReLU (replacing TLU) in the first layer is trained for comparison. Both models are also trained against the WOW steganography algorithm at the payload of 0.2 and 0.4 bpp. From Table 2, DFSE-Net with TLU has a lower error rate of detecting the WOW steganography algorithm, and the TLU function can also accelerate the convergence and show better performance, as shown in Figure 3.

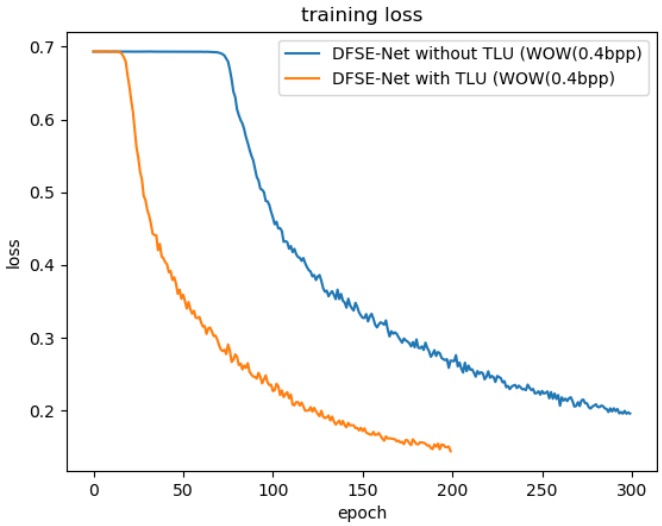

Figure 3. Comparing convergence performances of DFSE-Net with TLU and DFSE-Net without TLU (replaced with ReLU) against the WOW steganography algorithm at 0.4 bpp. Both models are trained and tested on the BOSS dataset.

Table 2. Steganalysis error rates comparison of DFSE-Net with truncated linear unit (TLU) and DFSE-Net with TeLU against the WOW steganography algorithm at 0.2 and 0.4 bpp. Both models are trained and tested on the BOSS dataset.

Algorithm	DFSE-Net with TLU	DFSE-Net with ReLU
WOW (0.2 bpp)	0.247	0.269
WOW (0.4 bpp)	0.149	0.171

3.1.5. Average Pooling Layer

The average pooling layer is used in each DFM. It calculates the average value of a certain area of the feature maps. It can reduce the size of feature maps according to the stride, reduce the parameters and calculation amount while retaining the main features. Furthermore, the average pooling layer can prevent over-fitting in training. Note that there is no pooling layer in the first block to avoid the loss of information as reported in [31]. Hence, we do not use the pooling layer after the first convolution layer.

3.2. Diverse Filter Module

As the Inception Network [24] in CNN gains a series of excellent results, it has been widely accepted that wider convolutional networks can capture more information of the images. Inspired by this, we designed the diverse filter modules called DFMs. They consisted of three different size convolutional kernels, as shown in Figure 4. As we can see, the types of kernel filters are 3×3, 5×5 and 1×1. The 3×3 and 5×5 convolutional kernels process the output of the previous layer parallelly. Then the outputs are concatenated together and sent to 1×1 convolutional layer. The 1×1 convolutional layer can effectively integrate the feature maps from above. In order to improve the performance, we take advantage of Xu-Net and Ye-Net to form DFMs using BN and ReLU layers. To further improve the effect, we design the SEMs to cooperate with DFMs.

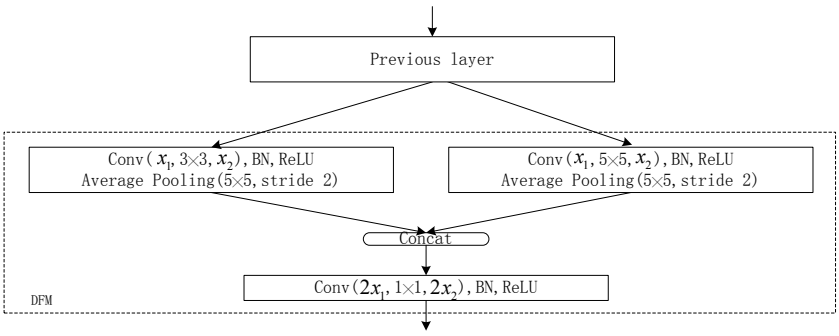

Figure 4. The design of the diverse filter module. The types and parameters are displayed inside boxes.

3.3. Squeeze-and-Excitation Module

The Squeeze-and-Excitation (SE) Module is not a complete network structure. It is a substructure that can be located in other classification or detection networks. In our architecture, each SEM is followed by the concatenated layer in the DFM, as shown in Figure 4. The core idea of SEM is to learn feature weights according to the loss in the training, so that the trained model can achieve better results in the way of effective feature maps with significant weights, invalid or ineffective feature maps with small weights. Therefore, the network can pay more attention to key channels. The overall design of SEM is presented in Figure 5.

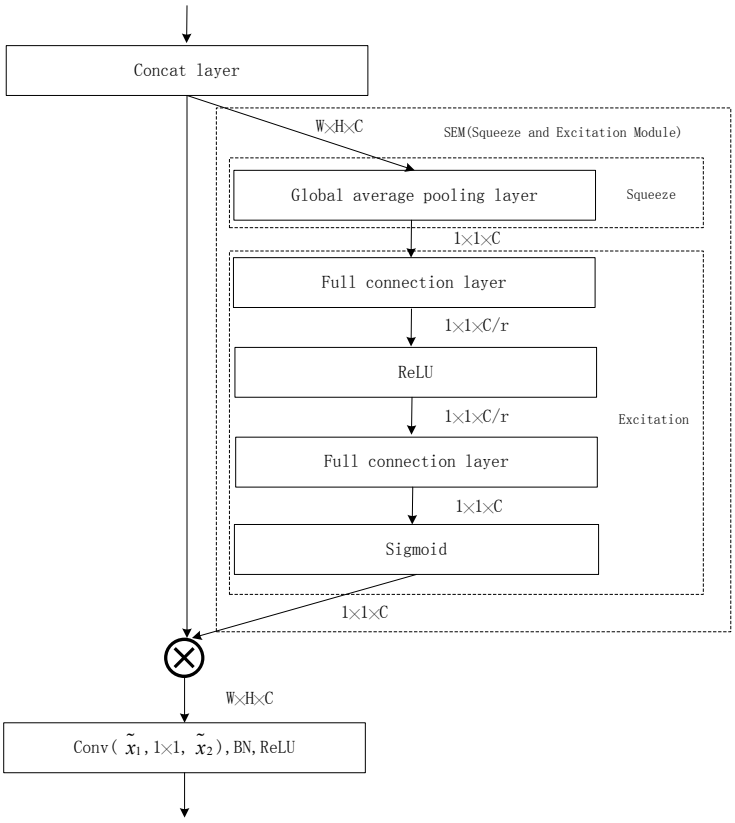

Figure 5. The design of the Squeeze-and-Excitation Module. The types of the layer are presented inside boxes and parameters are displayed outside boxes.

According to [25], the module can be divided into two steps: global information and recalibrated filter responses, also known as Squeeze-and-Excitation. In Figure 5, we know the main operations are global average pool (GAP)-full connection (FC)-ReLU-FC-Sigmoid. For the first step of squeeze, through the first global average pool layer, each output channel becomes a scalar. Therefore, the C channels will get the C scalars. As for the next step of excitation, by a set of operations of FC-ReLU-FC-Sigmoid, the C scalars will be normalized into [0, 1], as the channel weights. Finally, the operation of the scale rescales each channel by multiplying the weight, respectively.

To demonstrate the performance of SEM in image steganalysis, we compared them based on the DFSE network and without SEM after each DFM. Both models are also trained for the WOW steganography algorithm at 0.2 and 0.4 bpp. From Table 3, we can see DFSE-Net with SEM has a lower error rate of detecting the WOW steganography algorithm. SEM can also accelerate the convergence and show better performance, as shown in Figure 6.

Table 3. Steganalysis error rates comparison of DFSE-Net with SEM and DFSE-Net without SEM against the WOW steganography algorithm at 0.2 and 0.4 bpp. Both models are trained and tested on the BOSS dataset.

Algorithm	DFSE-Net with TLU	DFSE-Net with ReLU
WOW (0.2 bpp)	0.247	0.273
WOW (0.4 bpp)	0.149	0.170

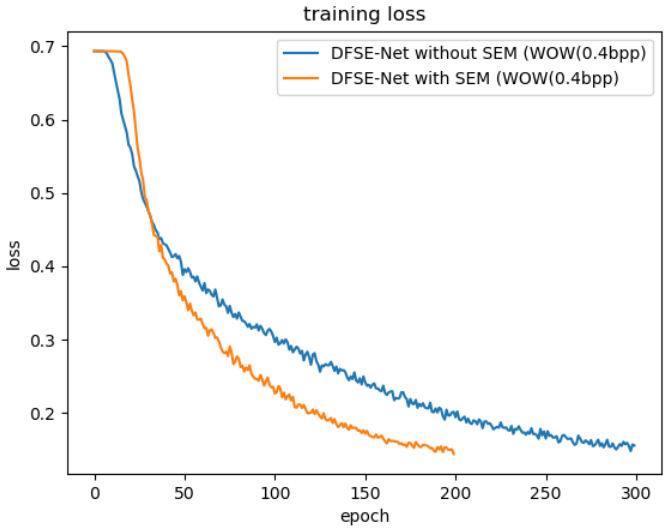

Figure 6. Comparing convergence performances of DFSE-Net with SEM and DFSE-Net without SEM against WOW steganography algorithm at 0.4 bpp. Both models are trained and tested on the BOSS dataset.

3.3.1. Squeeze

In order to exploit the channel dependencies, we consider each channel in the output features, as each convolutional filter operates with a small region and each unit of the output is also unable to utilize contextual information outside of this field. To exploit the channel dependencies, we use a global average pooling layer to squeeze global spatial information into a channel scalar. Formally, the statistic scalar z is generated by squeezing the input features U through its spatial dimensions $H \times W$, such that the c-th element of z is calculated by Equation (3):

$$z_c = F_{sq}(u_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i,j) \tag{3}$$

3.3.2. Excitation

To make full use of the information aggregated in the squeeze step, the excitation step is followed, which can capture channel-wise dependencies. To achieve this objective, the excitation step has to meet the following criteria. First, it must be able to learn the nonlinear interaction between channels. Second, it must learn a non-mutually-exclusive relationship. To meet these criteria, the sigmoid activation is employed. The operations of the excitation step can be formulated as Equation (4):

$$x_c = F_{ex}(z, W) = \sigma(g(z, W)) = \sigma(W_2 \delta(W_1 z)) \tag{4}$$

where W_1 refers to the first FC operation, δ refers to the ReLU function [32], W_2 refers to the second FC operation and σ refers to the sigmoid function.

To limit the complexity of the module, there is a parameter of reduction ratio r in the first FC layer to reduce the dimension of the input. Then the dimensionality is increased to the original channel dimension after the second FC layer. The final output $\hat{x}$ of the module is rescaled with the sigmoid activations s. The operation of the scale step can be formulated as Equation (5):

$$\hat{x}_c = F_{scale}(u_c, s_c) = s_c u_c \tag{5}$$

where $F_scale(uc; sc)$ refers to the multiplication operation between the scalar s_c and the corresponding feature map u_c and $\hat{x}_c$ is one of the channels of outputs $\hat{X}$.

To investigate the impact of parameter r in our network, we conduct several experiments with DFSE-Net for a range of different r values. The comparison results in Table 4 show the performance at each reduction ratio. There is only a slight difference in the detection accuracy. The set of $r = 6$ achieves a good balance between complexity and accuracy.

Table 4. Steganalysis error rates comparison of DFSE-Net against WOW steganography algorithm at 0.4 bpp at different reduction ratios. Both models are trained and tested on the BOSS dataset.

Ratio *r*	**1**	**2**	**4**	**6**	**8**	**15**
DFSE-Net	0.149	0.148	0.148	0.147	0.148	0.153

4. Experiments

In this section, several experiments are carried out to demonstrate the feasibility and effectiveness of DFSE-Net. For fair comparison, all methods are trained and tested on the same data sets.

4.1. The Steganographic Schemes and Datasets

In this paper, two content-adaptive image steganographic algorithms in the spatial domain of SUNIWARD and WOW were employed to product standard data sets. The two steganographic schemes were implemented with an STC simulator and the code files are available at http://dde.binghamton.edu/download/. In addition, the image sources of BOSSbase 1.01 can be found at the same URL. The image source is widely used in research fields, such as information hiding, forensics and steganalysis. It contains 10,000 8-bit grayscale images with a size of 512×512.

In consideration of the GPU computing power in our lab, the experiments on cover images with a size of 512×512 can be extremely time-consuming. Therefore, we decided to evaluate the effectiveness of all methods on the images with a size of 256×256. To this end, we refer to other models [20,23,33] and adopt the same approach. Therefore, we resampled all the images from 512×512 to 256×256 using the Matlab function of *imresize*() with default parameters.

Then, all 256×256 BOSSBase 1.01 images were embedded with messages using SUNIWARD and WOW steganographic algorithms, respectively, with the payload of 0.1, 0.2, 0.3, 0.4 and 0.5 bpp to generate the stego data sets. Therefore, we were able to generate 10 different steganographic data sets. Finally, all data sets including the cover set were split into three different sets randomly, 40% of the cover/stego pairs were split into the training set, 10% were split into the validation set, the rest were split into the testing set, and the testing set was untouched during all of the training phase.

4.2. Hyper-Parameters

We used Keras v2.24 with the backend of Tensorflow v1.15.3 for implementation. The optimizer of stochastic gradient descent (SGD) was applied to train our model. The momentum was set to 0.9 and the weight decay was fixed to 0.0001. No regularization and dropout were used. The batch size was fixed to 50 with 25 cover/stego pairs in the training procedure. For the preprocessing layer, thirty high-pass SRM filters were used without normalization. As the first layer, the TLU activation function was used and the threshold was set to three. All convolutional layers used the '*glorot_normal*' normal distribution initializer, also called the Xavier method. The fully-connected and softmax layers were initialized with the '*RandomNormal*' method of zero mean and standard deviation 0.01 and the initial bias was set to be zero. In addition to the above settings, the loss of our network was to minimize the cross-entropy. During the training phase, we set the maximum epoch of 500. Nevertheless, we usually cut short the training phase most of the time when the over-fitting phenomenon appeared. The learning rate (lr) was initialized to 0.01 and when the *val_loss* failed to improve after 10 epochs, the lr dropped by 10%. The minimum value of lr is 0.00001.

The performance was measured with the whole classification error probability on the same testing set using the formula $P_E = min_{P_{FA}} 1/2(P_{FA} + P_{MD})$, where P_{FA} and P_{MD} represent the probabilities of false-alarm and missed-detection.

4.3. Results

In this subsection, the experimental results are presented to verify the feasibility and demonstrate the effectiveness of our method. For fair comparison, we conducted all the experiments on the same data sets generated in Section 4.1, and the data sets in this paper are divided as follows. The 10,000 256 × 256 BOSSBase images were randomly split into three sets. The training set contains 4000 cover/stego image pairs, the validation set contains 1000 image pairs, and the testing set contains the remaining 5000 image pairs.

4.3.1. Feasibility

We have proved the validity of DFSE-Net on the data sets generated in Section 4.1, and the experimental results are shown in Figure 7 and Table 5. In Figure 7, we can see the DFSE-Net converges quickly on the two steganographic algorithms of WOW and S-UNIWARD at 0.4 bpp payload. According to Table 5, we can know the detection performance of DFSE-Net with different steganography methods at different payloads, and the experimental results show that it can detect stego images effectively.

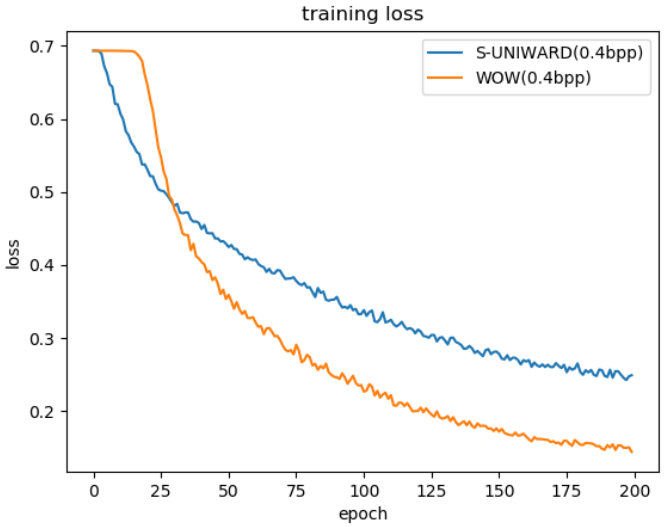

Figure 7. Comparing convergence performances of DFSE-Net against WOW and S-UNIWARD steganography algorithms at 0.4 bpp. Both models are trained and tested on the BOSS dataset.

Table 5. Steganalysis error rates of our method against WOW and S-UNIWARD algorithms at a range of different payloads from 0.1 to 0.5 bpp.

Algorithms	WOW	S-UNIWARN
0.1	0.342	0.422
0.2	0.247	0.341
0.3	0.193	0.284
0.4	0.149	0.215
0.5	0.124	0.189

4.3.2. Comparison with Existing Methods

To verify the superiority of our method, we conducted experiments to compare with the state-of-the-art approaches of the traditional classical method with the Spatial-Rich-Model (SRM) [13] combinined of Ensemble Classifier (EC), Xu-Net [20] and Ye-Net [18] without the selection-channel information (also called TLU-CNN), and Yedroudj-Net [20].

As the methods above are the current typical approaches. All methods have been trained and tested on the same datasets and run on a Nvidia P5000 GPU card.

In Table 6, we recorded the P_E compared with other state-of-the-art steganalyzers, and all the methods are compared against the steganographic algorithms WOW and S-UNIWARD, respectively, with the payload of 0.2 and 0.4 bpp.

Table 6. Steganalysis error rates comparison of the five steganalysis methods against two WOW and S-UNIWARD algorithms at 0.2 and 0.4 bpp.

	WOW		**S-UNIWARD**	
	0.2 bpp	**0.4 bpp**	**0.2 bpp**	**0.4 bpp**
SRM+EC	0.332	0.241	0.346	0.234
Xu-Net	0.345	0.245	0.389	0.277
Ye-Net	0.306	0.218	0.383	0.273
Yedroudj-Net	0.332	0.202	0.362	0.247
DFSE-Net	0.247	0.149	0.341	0.215

From Table 6, we can see that our method has better detection performance than other methods in terms of the steganographic algorithms WOW and S-UNIWARD, respectively, at payloads of 0.2 and 0.4 bpp. Since we have well designed DFSE-Net with DFMs and SEMs, the error rate of our proposed architecture is reduced by 8.5% compared with the traditional method of SRM+EC, by 6.7% compared with the Xu-Net, by 3.9% compared with the Ye-Net and by 3% compared with the Yedroudj-Net against WOW at 0.2 bpp. The results in Table 6 also show that our proposed network can effectively extract image features and classify input images.

As shown in Figure 8, we can see more intuitively that our proposed network has better performance than other methods on different steganographic algorithms at different payloads. The good performance also demonstrates the effectiveness of the network structure of DFMs and SEMs.

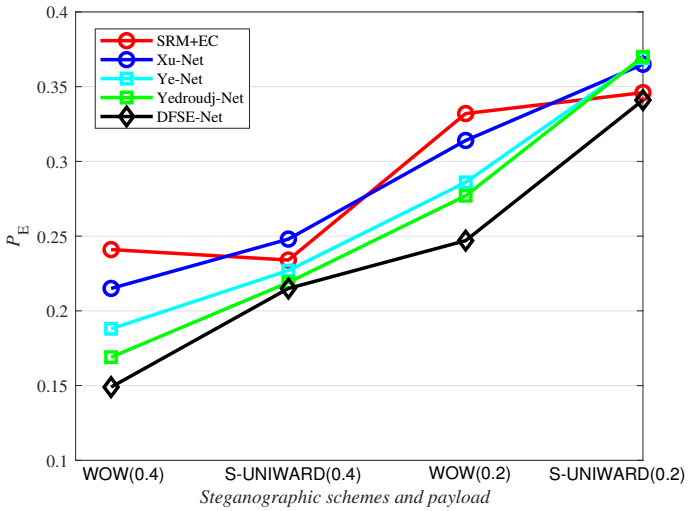

Figure 8. P_E comparison of the five steganalysis methods against WOW and S-UNIWARD algorithms with the payload of 0.2 and 0.4 bpp.

5. Conclusions

This paper presents the architecture of DFSE-Net with a carefully designed modules of diverse filters and Squeeze-and-Excitation for image steganalysis. DFSE-Net has gathered several latest design propositions, such as ABS, BN, TLU to build an efficient architecture beating the state-of-the-art methods. The experiments show that the P_E has reduced by 8.5% compared with the traditional method of SRM+EC, by 6.7% compared with the

Xu-Net, by 3.9% compared with the Ye-Net and by 3% compared with the Yedroudj-Net against WOW at 0.2 bpp. To summarize, the contributions of our method are reflected in two aspects: (i) proposing DFMs to capture more steganographic traces in a diverse way; (ii) proposing SEMs to enhance the effective features obtained from DFMs. Several experiments demonstrate the effectiveness and better performance of our method. There are also some limitations to our work. For example, our network can only deal with input images of the same size, while the images are in all sizes in real life. In the future, we consider adding more diverse structures to improve the detection efficiency and adding new modules to handle multi-size images.

Author Contributions: Conceptualization, F.L. and X.Y.; Data curation, X.Z. and Y.L.; Writing—review & editing, S.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China (Grant Number: 61602491).

Acknowledgments: The authors would like to thank the editor and the anonymous reviewers for their valuable comments.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Pevnỳ, T.; Filler, T.; Bas, P. Using high-dimensional image models to perform highly undetectable steganography. In *International Workshop on Information Hiding*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 161–177.
2. Holub, V.; Fridrich, J.; Denemark, T. Universal distortion function for steganography in an arbitrary domain. *Eurasip J. Inf. Secur.* **2014**, *2014*, 1. [CrossRef]
3. Holub, V.; Fridrich, J. Designing Steganographic Distortion Using Directional Filters. In Proceedings of the IEEE International Workshop on Information Forensics and Security (WIFS), Tenerife, Spain, 2–5 December 2012.
4. Li, B.; Wang, M.; Huang, J.; Li, X. A new cost function for spatial image steganography. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Paris, France, 27–30 October 2014; pp. 4206–4210.
5. Sedighi, V.; Cogranne, R.; Fridrich, J. Content-adaptive steganography by minimizing statistical detectability. *IEEE Trans. Inf. Forensics Secur.* **2015**, *11*, 221–234. [CrossRef]
6. Holub, V.; Fridrich, J. Digital image steganography using universal distortion. In Proceedings of the First ACM Workshop on Information Hiding and Multimedia Security, Montpellier, France, 17–19 June 2013; pp. 59–68.
7. Guo, L.; Ni, J.; Su, W.; Tang, C.; Shi, Y. Using Statistical Image Model for JPEG Steganography: Uniform Embedding Revisited. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 2669–2680. [CrossRef]
8. Kouider, S.; Chaumont, M.; Puech, W. Adaptive steganography by oracle (ASO). In Proceedings of the IEEE International Conference on Multimedia and Expo, San Jose, CA, USA, 15–19 July 2013.
9. Zhang, X.; Wang, S. Efficient steganographic embedding by exploiting modification direction. *IEEE Commun. Lett.* **2006**, *10*, 781–783. [CrossRef]
10. Yan, X.; Lu, Y.; Liu, L.; Song, X. Reversible image secret sharing. *IEEE Trans. Inf. Forensics Secur.* **2020**, *15*, 3848–3858. [CrossRef]
11. Wang, Z.; Zhang, X.; Yin, Z. Hybrid distortion function for JPEG steganography. *J. Electron. Imaging* **2016**, *25*, 050501. [CrossRef]
12. Liu, F.; Yan, X.; Lu, Y. Feature Selection for Image Steganalysis Using Binary Bat Algorithm. *IEEE Access* **2019**, *8*, 4244–4249. [CrossRef]
13. Pevny, T.; Bas, P.; Fridrich, J. Steganalysis by Subtractive Pixel Adjacency Matrix. *IEEE Trans. Inf. Forensics Secur.* **2010**, *5*, 215–224. [CrossRef]
14. Fridrich, J.; Kodovsky, J. Rich Models for Steganalysis of Digital Images. *IEEE Trans. Inf. Forensics Secur.* **2012**, *7*, 868–882. [CrossRef]
15. Holub, V.; Fridrich, J. Low-Complexity Features for JPEG Steganalysis Using Undecimated DCT. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 219–228. [CrossRef]
16. Denemark, T.; Sedighi, V.; Holub, V.; Cogranne, R.; Fridrich, J. Selection-channel-aware rich model for steganalysis of digital images. In Proceedings of the IEEE International Workshop on Information Forensics and Security (WIFS), Atlanta, GA, USA, 3–5 December 2014; pp. 48–53.
17. Kodovskỳ, J.; Fridrich, J. Calibration revisited. In Proceedings of the 11th ACM Workshop on Multimedia and Security, Princeton, NJ, USA, September 2009; pp. 63–74.
18. Qian, Y.; Dong, J.; Wang, W.; Tan, T. Deep learning for steganalysis via convolutional neural networks. *Proc. SPIE Int. Soc. Opt.* **2015**, *9409*, 94090J.
19. Xu, G.; Wu, H.Z.; Shi, Y.Q. Structural Design of Convolutional Neural Networks for Steganalysis. *IEEE Signal Process. Lett.* **2016**, *23*, 708–712. [CrossRef]

20. Ye, J.; Ni, J.; Yi, Y. Deep learning hierarchical representations for image steganalysis. *IEEE Trans. Inf. Forensics Secur.* **2017**, *12*, 2545–2557. [CrossRef]
21. Li, B.; Wei, W.; Ferreira, A.; Tan, S. ReST-Net: Diverse activation modules and parallel subnets-based CNN for spatial image steganalysis. *IEEE Signal Process. Lett.* **2018**, *25*, 650–654. [CrossRef]
22. Boroumand, M.; Chen, M.; Fridrich, J. Deep residual network for steganalysis of digital images. *IEEE Trans. Inf. Forensics Secur.* **2018**, *14*, 1181–1193. [CrossRef]
23. Yedroudj, M.; Comby, F.; Chaumont, M. Yedroudj-net: An efficient CNN for spatial steganalysis. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 2092–2096.
24. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
25. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7132–7141.
26. Bas, P.; Filler, T.; Pevnỳ, T. "Break Our Steganographic System": The Ins and Outs of Organizing BOSS. In *International Workshop on Information Hiding*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 59–70.
27. Masci, J.; Meier, U.; Cireşan, D.; Schmidhuber, J. Stacked convolutional auto-encoders for hierarchical feature extraction. In *International Conference on Artificial Neural Networks*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 52–59.
28. Kodovsky, J.; Fridrich, J.; Holub, V. Ensemble classifiers for steganalysis of digital media. *IEEE Trans. Inf. Forensics Secur.* **2011**, *7*, 432–444. [CrossRef]
29. 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.* **2019**, *14*, 2735–2748. [CrossRef]
30. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv* **2015**, arXiv:1502.03167.
31. Pibre, L.; Pasquet, J.; Ienco, D.; Chaumont, M. Deep learning is a good steganalysis tool when embedding key is reused for different images, even if there is a cover sourcemismatch. In Proceedings of the IST International Symposium on Electronic Imaging, San Francisco, CA, USA, 14–18 Febuary 2016.
32. Nair, V.; Hinton, G.E. Rectified Linear Units Improve Restricted Boltzmann Machines. In Proceedings of the 27th International Conference on Machine Learning, Haifa, Israel, 21 June 2010; pp. 807–814.
33. Zhang, R.; Zhu, F.; Liu, J.; Liu, G. Depth-Wise Separable Convolutions and Multi-Level Pooling for an Efficient Spatial CNN-Based Steganalysis. *IEEE Trans. Inf. Forensics Secur.* **2019**, *15*, 1138–1150. [CrossRef]